ROHM GROUP
LAPIS SEMICONDUCTOR

ML630Q791
User's Manual

Issue Date: Oct.22, 2014

ROHM SEMICONDUCTOR

Notes

1) The information contained herein is subject to change without notice.
2) Although LAPIS Semiconductor is continuously working to improve product reliability and quality, semiconductors can break down and malfunction due to various factors. Therefore, in order to prevent personal injury or fire arising from failure, please take safety measures such as complying with the derating characteristics, implementing redundant and fire prevention designs, and utilizing backups and fail-safe procedures. LAPIS Semiconductor shall have no responsibility for any damages arising out of the use of our Products beyond the rating specified by LAPIS Semiconductor.
3) Examples of application circuits, circuit constants and any other information contained herein are provided only to illustrate the standard usage and operations of the Products.The peripheral conditions must be taken into account when designing circuits for mass production.
4) The technical information specified herein is intended only to show the typical functions of the Products and examples of application circuits for the Products. No license, expressly or implied, is granted hereby under any intellectual property rights or other rights of LAPIS Semiconductor or any third party with respect to the information contained in this document; therefore LAPIS Semiconductor shall have no responsibility whatsoever for any dispute, concerning such rights owned by third parties, arising out of the use of such technical information.
5) The Products are intended for use in general electronic equipment (i.e. AV/OA devices, communication, consumer systems, gaming/entertainment sets) as well as the applications indicated in this document.
6) The Products specified in this document are not designed to be radiation tolerant.
7) For use of our Products in applications requiring a high degree of reliability (as exemplified below), please contact and consult with a LAPIS Semiconductor representative: transportation equipment (i.e. cars, ships, trains), primary communication equipment, traffic lights, fire/crime prevention, safety equipment, medical systems, servers, solar cells, and power transmission systems.
8) Do not use our Products in applications requiring extremely high reliability, such as aerospace equipment, nuclear power control systems, and submarine repeaters.
9) LAPIS Semiconductor shall have no responsibility for any damages or injury arising from non-compliance with the recommended usage conditions and specifications contained herein.
10) LAPIS Semiconductor has used reasonable care to ensure the accuracy of the information contained in this document. However, LAPIS Semiconductor does not warrant that such information is error-free and LAPIS Semiconductor shall have no responsibility for any damages arising from any inaccuracy or misprint of such information.
11) Please use the Products in accordance with any applicable environmental laws and regulations, such as the RoHS Directive. For more details, including RoHS compatibility, please contact a ROHM sales office. LAPIS Semiconductor shall have no responsibility for any damages or losses resulting non-compliance with any applicable laws or regulations.
12) When providing our Products and technologies contained in this document to other countries, you must abide by the procedures and provisions stipulated in all applicable export laws and regulations, including without limitation the US Export Administration Regulations and the Foreign Exchange and Foreign Trade Act.
13) This document, in part or in whole, may not be reprinted or reproduced without prior consent of LAPIS Semiconductor.

Copyright 2014 LAPIS Semiconductor Co., Ltd.

LAPIS Semiconductor Co.,Ltd.

.

2-4-8 Shinyokohama, Kouhoku-ku,
Yokohama 222-8575, Japan
http://www.lapis-semi.com/en/

Preface

This manual describes the hardware and operation of the ML630Q791 32-bit microcontrollers.

Please ensure that you refer to the latest versions.

The following manuals are also provided. Read them as necessary.

- Cortex®-M0 Technical Reference Manual (DDI0432)
- Cortex®-M0 Devices Generic User Guide (DUI0497)

The documents above are published by ARM Limited. Please ensure that you refer to the latest versions.

ARM, Cortex, Thmub are registered trademarks of ARM Limited in the EU and other countries.

Notation

Classification	Notation	Description
♦ Numeric value	0xnn, 0xnnnn_nnnn	Indicates a hexadecimal number.
	0bnn, 0bnnnn_nnnn	Indicates a binary number.
♦ Unit	word, W	1 word = 16 bits
	byte, B	1 byte = 8 bits
	nibble, N	1 nibble = 4 bits
	maga-, M	10^6
	kilo-, K	$2^{10} = 1024$
	kilo-, k	$10^3 = 1000$
	milli-, m	10^{-3}
	micro-, μ	10^{-6}
	nano-, n	10^{-9}
	second, s (lower case)	second
♦ Terminology	"H" level, "1" level	Indicates high voltage signal levels V_{IH} and V_{OH} as specified by the electrical characteristics.
	"L" level, "0" level	Indicates low voltage signal levels V_{IL} and V_{OL} as specified by the electrical characteristics.

♦ Register description

R/W: Indicates that Read/Write attribute. "R" indicates that data can be read and "W" indicates that data can be written. "R/W" indicates that data can be read or written.

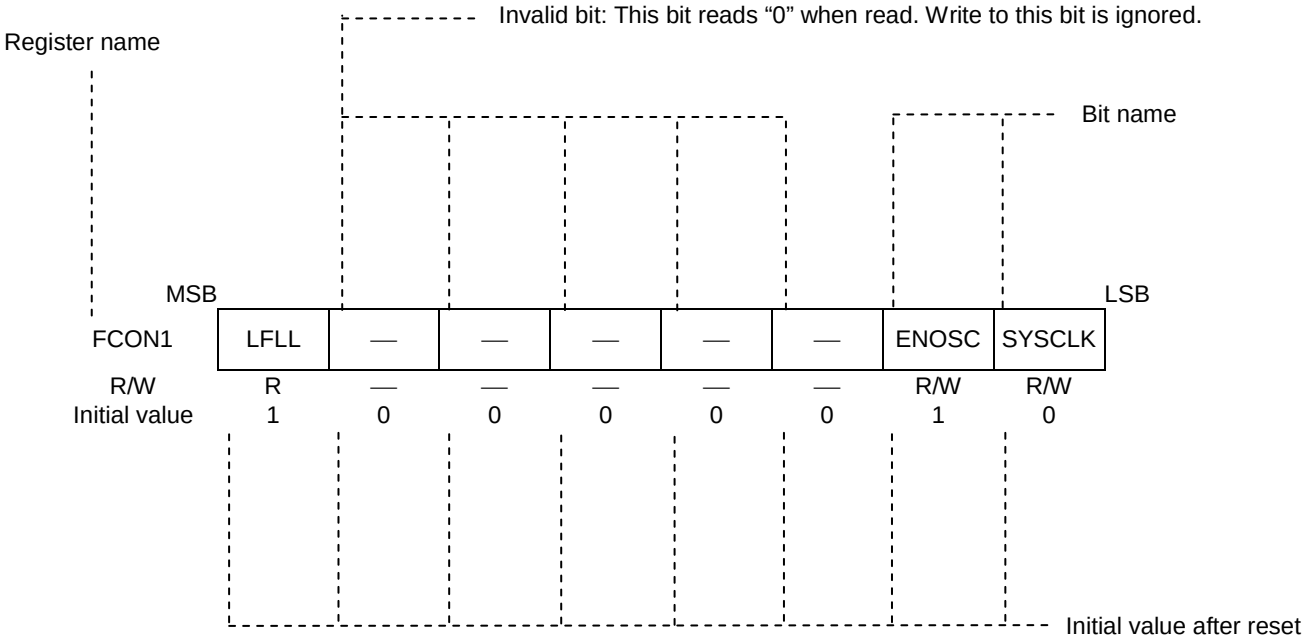


Table of Contents

Chapter 1

1. Overview 1-1
1.1. Features 1-1
1.2. Configuration of Function Blocks 1-3
1.2.1. ML630Q791 Block Diagram 1-3
1.3. Pin Layout 1-4
1.3.1. Pin Layout 1-4
1.3.1.1. WL-CSP Package 1-4
1.4. List of Pins 1-5
1.4.1. Pin Description 1-6
1.4.2. Handling of Unused Pins 1-8

Chapter 2

2. CPU 2-1
2.1. Overview 2-1
2.1.1. Features 2-1
2.2. Description of Registers 2-2
2.2.1. List of Registers 2-2

Chapter 3

3. Memory Space 3-1
3.1. Overview 3-1
3.2. Memory Map 3-1
3.3. Internal Memory 3-4
3.3.1. Internal Flash ROM 3-4
3.3.2. Work RAM 3-4
3.4. Memory Controller Function 3-5
3.4.1. List of Registers 3-5
3.4.2. Remapping Control Register (SYSCON_REMAP_CON) 3-6
3.4.3. Remapping Base Address Register (SYSCON_REMAP_BASE) 3-7
3.4.4. Boot/Remapping Function 3-8
3.5. Access Response for Memory Space 3-8

Chapter 4

4. Reset Functions 4-1
4.1. Overview 4-1
4.1.1. Features 4-1
4.1.2. Configuration 4-1
4.1.3. List of Pins 4-1
4.2. Description of Registers 4-2
4.3. Description of Operation 4-3
4.3.1. System Reset Mode 4-3

Chapter 5

5. MCU Control Function 5-1
5.1. Overview 5-1
5.1.1. Features 5-1
5.2. Description of Registers 5-2
5.2.1. List of Registers 5-2
5.2.2. Revision Register (IDR) 5-3
5.2.3. Peripheral Clock Enable Register (PECLKEN) 5-4
5.2.4. Peripheral Clock Disable Register (PECLKDIS) 5-5
5.2.5. Peripheral Reset Enable Register (PERSTEN) 5-6
5.2.6. Peripheral Reset Disable Register (PERSTDIS) 5-7
5.2.7. Part A Mode Setting Register (PAMOD) 5-8
5.3. Description of Operation 5-10
5.3.1. State Transition 5-10
5.3.2. State of Each Operation Mode 5-11

Chapter 6

6. Clock 6-1
6.1. Overview 6-1
6.1.1. Features 6-1
6.1.2. Configuration 6-1
6.2. Description of Registers 6-2
6.2.1. List of Registers 6-2
6.2.2. Frequency Control Register 0 (FCON0) 6-3
6.2.3. Frequency Control Register 1 (FCON1) 6-4
6.2.4. High-Speed Time Base Counter Frequency Divide Register (HTBDR) 6-5
6.3. Description of Operation 6-6
6.3.1. Low-Speed Clock 6-6
6.3.2. High-Speed Clock 6-6
6.3.2.1. Internal FLL Oscillation 6-6
6.3.3. Low-Speed Time Base Counter 6-6
6.3.4. High-Speed Time Base Counter 6-6

Chapter 7

7. Interrupt 7-1
7.1. Overview 7-1
7.1.1. Features 7-1
7.2. Descriptoon of Registers 7-2
7.2.1. List of Registers 7-2
7.2.2. Correspondence with Bits 7-3
7.3. Descrtption of Operation 7-4

Chapter 8

8. Timer 8-1
8.1. Overview 8-1
8.1.1. Features 8-1
8.1.2. Configuration 8-1
8.1.3. List of Pins 8-2
8.2. Description of Registers 8-3
8.2.1. List of Registers 8-3
8.2.2. Timer n Data Register (TMnD: n = 0, 2, 4, 6) 8-4
8.2.3. Timer m Data Register (TMmD: m = 1, 3, 5, 7) 8-5
8.2.4. Timer n Counter Register (TMnC: n = 0, 2, 4, 6) 8-6
8.2.5. Timer m Counter Register (TMmC: m = 1, 3, 5, 7) 8-7
8.2.6. Timer n Control Register 0 (TMnCON0: m = 1, 3, 5, 7) 8-8
8.2.7. Timer m Control Register 0 (TMmCON0: m = 1, 3, 5, 7) 8-9

8.2.8. Timer n Control Register 1 (TMnCON1: n = 0, 2, 4, 6) 8-10
8.2.9. Timer m Control Register 1 (TMmCON1: m = 1, 3, 5, 7) 8-11
8.3. Description of Operation 8-12
8.3.1. Timer Mode Operation 8-12
8.4. Specifying Port Registers 8-13
8.4.1. Operating Timer with External Clock (PWM) 8-13

Chapter 9

9. PWM 9-1
9.1. Overview 9-1
9.1.1. Features 9-1
9.1.2. Configuration 9-1
9.1.3. List of Pins 9-2
9.2. Description of Registers 9-2
9.2.1. List of Registers 9-2
9.2.2. PWM0 Cycle Register (PW0P) 9-3
9.2.3. PWM0 Duty Register (PW0D) 9-4
9.2.4. PWM0 Counter Register (PW0C) 9-5
9.2.5. PWM0 Control Register 0 (PW0CON0) 9-6
9.2.6. PWM0 Control Register 1 (PW0CON1) 9-7
9.3. Description of Operation 9-8
9.4. Port Register Settings 9-10
9.4.1. Functioning PA0 Pin (PWM0) as PWM Output 9-10
9.4.2. Operating PWM0 with External Clock (PWM) 9-11

Chapter 10

10. Watchdog Timer 10-1
10.1. Overview 10-1
10.1.1. Features 10-1
10.1.2. Configuration 10-1
10.2. Description of Registers 10-2
10.2.1. List of Registers 10-2
10.2.2. Watchdog Timer Control Register (WDTCON) 10-3
10.2.3. Watchdog Timer Mode Register (WDTMOD) 10-4
10.3. Description of Operation 10-5
10.4. Example of Processing When Not Using Watchdog Timer 10-7

Chapter 11

11. Host Interface 11-1
11.1. Overview 11-1
11.1.1. Features 11-1
11.1.2. Configuration 11-1
11.2. Description of Registers 11-2
11.2.1. Register List (CPURG : for CPU Access) 11-2
11.2.2. Register List (HSTRG : for Host Access) 11-4
11.2.3. CPURG : Configuration Register (HIFCFG) 11-6
11.2.4. CPURG : Operation Status Register (HIFST) 11-8
11.2.5. CPURG : Interrupt Request Register (HIFRQ) 11-9
11.2.6. CPURG : FIFO Register (HIFFIFO) 11-10
11.2.7. CPURG : FIFO Switch Register (HIFFSEL) 11-11
11.2.8. CPURG : FIFO Write Pointer Register (HIFWP) 11-12
11.2.9. CPURG : FIFO Read Pointer Register (HIFRP) 11-13
11.2.10. CPURG : Parameter Register (HIFPRMF, HIFPRMB, HIFPRM7, HIFPRM3) 11-14
11.2.11. CPURG : Command Register (HIFCMD) 11-15
11.2.12. CPURG: Result Register n (HIFRLTn) n:00 to 3F 11-16
11.2.13. HSTRG : Configuration Register (CFG) 11-17
11.2.14. HSTRG : Interrupt Mask Register 0 (INTMSK0) 11-18

11.2.15.HSTRG : Interrupt Mask Register 1 (INTMSK1) 11-19
11.2.16.HSTRG : Operation Status Register (STATUS)....11-20
11.2.17.HSTRG : Error Code Register 0 (ERROR0) 11-21
11.2.18.HSTRG : Error Code Register 1 (ERROR1) 11-22
11.2.19.HSTRG : Interrupt Request Register 0 (INTREQ0)....11-23
11.2.20.HSTRG : Interrupt Request Register 1 (INTREQ1)....11-24
11.2.21.HSTRG : FIFO Register (FIFO)....11-25
11.2.22.HSTRG : Parameter Register n (PRMn) n:00 to 0F 11-26
11.2.23.HSTRG : Command Register 0n (CMDn) n:0 to 1 11-27
11.2.24.HSTRG : Command Entry Register (ENT)....11-28
11.2.25.HSTRG : Result Register n (RSLTn) n:00 to 3F....11-29
11.3. Serial Interface 11-30
11.3.1. I2C Interface....11-30
11.3.1.1. I2C Slave Interface....11-30
11.3.1.2. I2C Trsnsfer Format....11-31
11.3.2. SPI Interface 11-33
11.3.2.1. SPI Trsnsfer Format....11-34
11.4. Description of Operation 11-36
11.4.1. Power-down and Resume 11-36
11.4.2. Command Input and Processing of Response....11-36
11.4.3. Write/Read Data to/from FIFO....11-37
11.4.4. Register Access Conflict by Host and CPU....11-39
11.4.5. Timing of Clearing Interrupt Request Register....11-39
11.4.6. Error Handling....11-40
11.4.7. Error Status Notification....11-41
11.4.8. Clock Requirements....11-41
11.5. Specifying Port Registers 11-42
11.5.1. When Using SPI Interface (Three-wire)....11-42
11.5.2. When Using SPI Interface (Four-wire)....11-43
11.5.3. When INT0_S Is Used....11-44
11.5.4. When INT1_S Is Used....11-45

Chapter 12

12. I2C Bus Interface 12-1
12.1. Overview 12-1
12.1.1. Features.... 12-1
12.1.2. Configuration.... 12-1
12.1.3. List of Pins.... 12-1
12.2. Description of Registers 12-2
12.2.1. List of Registers.... 12-2
12.2.2. I2C Control Register (I2CCTL0, 1) 12-3
12.2.3. I2C Status Register (I2CSR0, 1).... 12-6
12.2.4. I2C Data Register (I2CDR0, 1) 12-10
12.2.5. I2C Bus Monitor Register (I2CMON0, 1)....12-11
12.2.6. I2C Bus Transfer Rate Setup Counter (I2CBC0, 1) 12-12
12.2.7. I2C Mode Register (I2CMOD0, 1)....12-13
12.2.8. I2C Buffer Mode Slave Address Register (I2CBUFSLV0, 1) 12-14
12.2.9. I2C Buffer Mode Sub Address Register (I2CBUFSUB0, 1)....12-15
12.2.10.I2C Buffer Mode Format Register (I2CBUFFOR0, 1)....12-16
12.2.11.I2C Buffer Mode Control Register (I2CBUFCTL0, 1) 12-17
12.2.12.I2C Buffer Mode Interrupt Mask Register (I2CBUFMSK0, 1) 12-18
12.2.13.I2C Buffer Mode Status Register (I2CBUFSTA0, 1)....12-20
12.2.14.I2C Buffer Mode Level Register (I2CBUFLEV0, 1)....12-22
12.2.15.I2C Timer Register (I2CTMR0, 1)....12-23
12.2.16.I2C Input Noise Filter Setting Register (I2CNF0, 1)....12-24
12.3. Description of Operation 12-25
12.3.1. Flow of Initial Setting....12-25
12.3.2. Flow of Master Transmission 12-27
12.3.3. Flow of Master Reception....12-28

12.3.4. Flow of Compound Mode (Receiving by Master after Transmitting from Master)....12-29
12.3.5. Flow of Compound Mode (Transmitting from Master after Receiving by Master)....12-31
12.3.6. Flow When Using Buffer Mode....12-33
12.3.7. Flow of Mode Switching....12-35
12.3.7.1. Flow of Switching to Normal Mode....12-35
12.3.7.2. Flow of Switching to Buffer Mode....12-35
12.4. Waveform in Each Mode....12-36
12.4.1. Waveform Transmitted by Master....12-36
12.4.2. Waveform Received by Master....12-36
12.4.3. Waveform of Compound Format (Master Transmission + Master Reception)....12-36
12.4.4. Waveform of Compound Format (Master Reception + Master Transmission)....12-36
12.4.5. Waveform 1 When Using Buffer Mode....12-37
12.4.6. Waveform 2 When Using Buffer Mode....12-37
12.4.7. Waveform 3 When Using Buffer Mode....12-37
12.4.8. Waveform 4 When Using Buffer Mode....12-38
12.4.9. Waveform 5 When Using Buffer Mode....12-38
12.4.10. Waveform 6 When Using Buffer Mode....12-38
12.4.11. Waveform 7 When Using Buffer Mode....12-39
12.4.12. Waveform 8 When Using Buffer Mode....12-39
12.5. Restrictions....12-40
12.6. Specifying Port Registers....12-41
12.6.1. Operating I2C1....12-41

Chapter 13

13. UART....13-1
13.1. Overview....13-1
13.1.1. Features....13-1
13.1.2. Configuration....13-2
13.1.3. List of Pins....13-2
13.2. Description of Registers....13-3
13.2.1. List of Registers....13-3
13.2.2. UART Receive Data Register / UART Transmit Data Register / UART Baud Rate Dividing Register(LSB) (UARTRBR, UARTTHR, UARTDLL)....13-4
13.2.3. UART Interrupt Enable Register / UART Baud Rate Dividing Register(MSB) (UARTIER, UARTDLM)....13-5
13.2.4. UART Interrupt Status Register / UART FIFO Control Register (UARTIIR, UARTFCR)....13-7
13.2.5. UART Line Control Register (UARTLCR)....13-10
13.2.6. UART Line Status Register (UARTLSR)....13-12
13.2.7. UART Scratchpad Register (UARTSCR)....13-15
13.3. Description of Operation....13-16
13.3.1. Data Transmission....13-16
13.3.2. Data Reception....13-17
13.3.3. Baud Rate Clock Generation....13-19
13.3.4. FIFO Mode....13-20
13.3.5. FIFO Polled Mode....13-21
13.3.6. Error Status....13-22
13.3.7. Setting Example....13-23
13.4. Specifying Port Registers....13-24
13.4.1. Operating UART....13-24

Chapter 14

14. Arithmetic Circuit....14-1
14.1. Overview....14-1
14.1.1. Features....14-1
14.1.2. Configuration....14-1
14.2. Description of Registers....14-2
14.2.1. List of Registers....14-2
14.2.2. Operation Status Register (CALSTS)....14-3
14.2.3. Operation Input Register AL (CALAL)....14-4

14.2.4. Operation Input Register AH (CALAH) 14-5
14.2.5. Operation Input Register BL (CALBL) 14-6
14.2.6. Operation Input Register BH (CALBH) 14-7
14.2.7. Calculation Result Register 0L (CALR0L) 14-8
14.2.8. Calculation Result Register 0H (CALR0H) 14-9
14.2.9. Calculation Result Register 1L (CALR1L) 14-10
14.2.10. Calculation Result Register 1H (CALR1H) 14-11
14.3. Description of Operation 14-12
14.3.1. Division 14-12
14.3.2. Root Operation 14-12

Chapter 15

15. GPIO 15-1
15.1. Overview 15-1
15.1.1. Features 15-1
15.1.2. Configuration 15-1
15.1.3. List of Pins 15-1
15.2. Description of Registers 15-2
15.2.1. List of Registers 15-2
15.2.2. GPIO Port Data Register (PIODAT) 15-3
15.2.3. GPIO Port Direction Register (PIODIR) 15-4
15.2.4. GPIO Port Control Register (PIOCON) 15-5
15.2.5. GPIO Interrupt Enable Register (PIOIE) 15-6
15.2.6. GPIO Interrupt Mode Register (PIOIM) 15-7
15.2.7. GPIO Interrupt Status Register (PIOIS) 15-8
15.3. Description of Operation 15-9
15.3.1. I/O Setting 15-9
15.3.2. Interrupt Setting Procedure 15-9
15.3.3. Various Interrupt Operations 15-9
15.3.3.1. Falling Edge Interrupt Mode 15-10
15.3.3.2. Rising Edge Interrupt 15-11
15.3.3.3. L-level Input Interrupt 15-12
15.3.3.4. H-level Input Interrupt 15-13
15.3.3.5. Both-edge (Rising/Falling) Interrupt 15-14

Chapter 16

16. Flash Programming 16-1
16.1. General Description 16-1
16.1.1. Features 16-1
16.2. Description of Registers 16-2
16.2.1. List of Registers 16-2
16.2.2. Flash-ROM Status Register (FLCSTA) 16-3
16.2.3. Flash-ROM Acceptor Register (FLCACP) 16-4
16.2.4. Flash-ROM Address Register (FLCADR) 16-5
16.2.5. Flash-ROM Write Data Register (FLCWDA) 16-6
16.2.6. Flash-ROM Erase Register (FLCERA) 16-7
16.2.7. Flash-ROM Size Register (FLCRSIZ) 16-8
16.2.8. Boot Program Address Register (FLCBADR) 16-9
16.2.9. Set Count Acceptor Register (FLCSCACP) 16-10
16.2.10. Set Count Sector Erase CE Enable Register (FLCSCSCE) 16-11
16.2.11. Set Count Program CE Enable Register (FLCSCPCE) 16-12
16.2.12. Set Count Sector Erase WE Enable Register (FLCSCSWE) 16-13
16.2.13. Set Count Program WE Enable Register (FLCSCPWE) 16-14
16.2.14. Set Count Program PROG2_1st Enable Register (FLCSCPP21E) 16-15
16.2.15. Set Count Program PROG2_1st Disable Register (FLCSCPP21D) 16-16
16.2.16. Set Count Program PROG2_2nd Enable Register (FLCSCPP22E) 16-17
16.2.17. Set Count Program PROG2_2nd Disable Register (FLCSCPP22D) 16-18
16.2.18. Set Count Program PROG2_3rd Enable Register (FLCSCPP23E) 16-19

16.2.19.Set Count Program PROG2_3rd Disable Register (FLCSCPP23D) 16-20
16.2.20.Set Count Program PROG2_4th Enable Register (FLCSCPP24E) 16-21
16.2.21.Set Count Program PROG2_4th Disable Register (FLCSCPP24D)....16-22
16.2.22.Set Count Program BYTE1 Enable Register (FLCSCPB1E) 16-23
16.2.23.Set Count Program BYTE2 Enable Register (FLCSCPB2E) 16-24
16.2.24.Set Count Program BYTE3 Enable Register (FLCSCPB3E) 16-25
16.2.25.Set Count Erase WE Disable Register (FLCSCEWED)....16-26
16.2.26.Set Count Program WE Disable Register (FLCSCPWED) 16-27
16.2.27.Set Count CE Disable Register (FLCSCCED) 16-28
16.2.28.Set Count Termination Register (FLCSCEND)....16-29
16.3. Description of Operation 16-30
16.3.1. Erase/Write Flash-ROM 16-30
16.3.2. Counter Setting 16-30
16.3.3. Sector Erase 16-31
16.3.4. 1-word Write....16-32
16.3.5. Erase/Write to Area Where Flash-ROM Is Not Implemented 16-33
16.3.6. Notes in Use....16-33

Chapter 17

17. On-Chip Debug Functiion 17-1
17.1. Overview 17-1

Chapter 18

18. Power Supply Circuit 18-1
18.1. Overview 18-1
18.1.1. Features....18-1
18.1.2. Configuration....18-1
18.1.3. List of Pins....18-1

Appendixes

Appendix A Registers A-1
Appendix B Package Dimensions B-1
Appendix C Electrical Characteristics....C-1
Appendix D Application Circuit Example D-1

Revision History

Revision History....R-1

Chapter 1 Overview

1. Overview

1.1 Features

This LSI is a high-performance low power 32-bit microcontroller optimized for the control of various sensor ICs. Equipped with a 32-bit CPU core Cortex®-M0, it implements a 128 KB flash memory, 16 KB RAM, rich interfaces used to control various sensors, and host interface with the 512-byte communication register in a very compact package. This LSI can efficiently control the power consumption of the whole system by separating the sensor control function from the application processor, and its high performance permits sensor-fusion using accelerometers, magnetic field sensors and gyro sensors, which makes it an ideal sensor control microcontroller for smart phones.

- CPU
  - 32-bit RISC CPU (CPU name: ARM® Cortex®-M0)
  - Thumb®/Thumb®-2 instruction supported
  - Serial Wire Debug Port supported

- Internal memory
  - 128 KB FLASH ROM (32K x 32-bit)
  - 16 KB SRAM (4K x 32-bit)

- Interrupt controller
  - Non-maskable interrupt: 1 source
  - Maskable interrupt: 21 sources
    Number of internal sources: 14 (Timer: 8, PWM: 1, I2C: 2, HOSTIF: 1, Arithmetic circuit: 1, UART: 1)
    Number of external sources: 7

- Timer
  - 8 bit x 8 ch [also available is 16-bit configuration (using Timers 0 and 1, Timers 2 and 3, Timers 4 and 5, or Timers 6 and 7) x 4 ch]
  - Watchdog timer (WDT) x 1ch
  - 16-bit PWM x 1ch

- Serial interface
  - I2C interface with master function x 2ch (including 8-bit, 32-stage FIFO)
  - UART interface x 1ch (two-wire, full duplex buffer system, including 8-bit, 16-stage FIFO)

- Host interface
  - Serial interface with slave function (SPI/I2C selectable) x 1ch
  - 1ch host processor interrupt
  - 512-Byte FIFO RAM for communication

- General-purpose I/O port
  - 7-bit input/output port x 1ch
  - External interrupt input available

- Arithmetic circuit
  - Root operation
  - Division operation

- Flash rewrite function
  - Hardware remap supported
  - ISP supported

- Operation mode and power consumption control function
  - CPU operation mode
    Operation with a high-speed and low-speed clock are supported.
  - Sleep mode
    The sleep mode, which stops CPU only, is supported.
  - Sleep deep mode
    The sleep deep mode, which stops CPU and peripheral blocks, is supported.

- Input clock
  - 32.768 kHz (External clock input)

- Power supply voltage
  - V_{DD} : 1.7V to 1.9V
  - Digital core section : 1.35V to 1.65V (supplied by the internal voltage regulator)

- Supply current (Typ)
  - High-speed operation (32 MHz) : 5.0 mA
  - Low-speed operation (32.768 kHz) : 0.5 mA
  - sleepdeep mode : 2.5uA

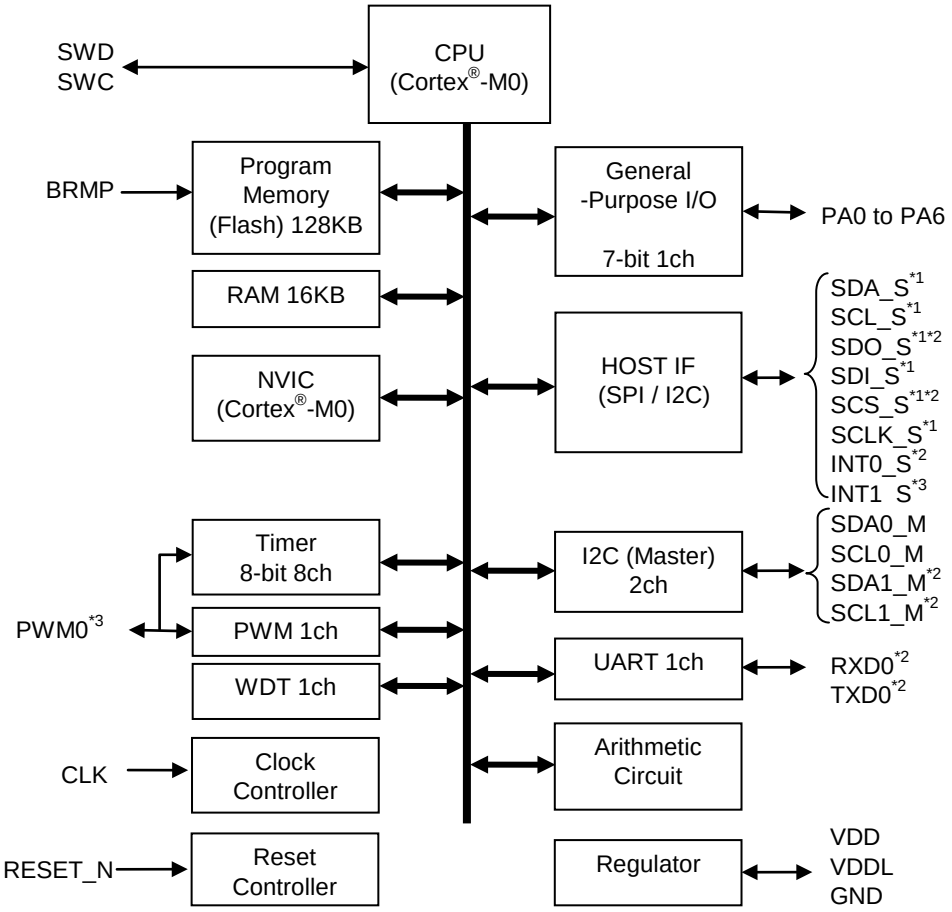
- Operating frequency
  - High-speed clock (FLL): 32 MHz
  - Low-speed clock: 32.768 kHz

- Operating temperature
  - -40°C to 85°C

- Package
  - 20-pin WL-CSP 0.4 mm pitch (2.1 mm x 1.8 mm)

1.2 Configuration of Function Blocks

1.2.1 ML630Q791 Block Diagram



*1 Selectable I2C or SPI interface
*2 Secondary function
*3 Tertiary function

Figure 1-1 Block Diagram of ML630Q791

1.3 Pin Layout

1.3.1 Pin Layout

1.3.1.1 WL-CSP Package

PA4	PA5	SDA0_M	SCL0_M	5
PA3	PA2	BRMP	VPP	4
GND	PA0	SWD	SWC	3
VDDL	PA1	SDA_S	CLK	2
VDD	RESET_N	PA6	SCL_S	1
D	C	B	A	

20-pin WL-CSP Package (S-UFLGA20-1.84x2.14-0.40-W)
(Bottom View)

Figure 1-2 Pin Layout of ML630Q791 Package

1.4 List of Pins

PIN No.	Primary function				Secondary function			Tertiary function		
	Pin name	I/O	Reset state	Description	Pin name	I/O	Description	Pin name	I/O	Description
D3	GND	—	—	Power Supply	—	—	—	—	—	—
D1	VDD	—	—	Power Supply	—	—	—	—	—	—
D2	VDDL	—	—	Power Supply	—	—	—	—	—	—
A2	CLK	I	HZ	SYSTEM	—	—	—	—	—	—
B4	BRMP	I	PD	SYSTEM	—	—	—	—	—	—
A3	SWC	I	PU	DEBUG I/F	—	—	—	—	—	—
B3	SWD	IO	PU	DEBUG I/F	—	—	—	—	—	—
C1	RESET_N	I	PU	SYSTEM	—	—	—	—	—	—
A1	SCL_S*1 SCLK_S	I I	HZ	HSTIF	—	—	—	—	—	—
B2	SDA_S*1 SDIO_S SDI_S	IO IO I	HZ	HSTIF	—	—	—	—	—	—
D5	PA4	IO	HZ	GPIO	SCS_S	I	HSTIF	—	—	—
C5	PA5	IO	HZ	GPIO	SDO_S	O	HSTIF	—	—	—
B5	SDA0_M	IO	HZ	I2C0	—	—	—	—	—	—
A5	SCL0_M	O	HZ	I2C0	—	—	—	—	—	—
C4	PA2	IO	HZ	GPIO	RXD0	I	UART	—	—	—
D4	PA3	IO	HZ	GPIO	TXD0	O	UART	—	—	—
C3	PA0	IO	HZ	GPIO	SDA1_M	IO	I2C1	PWM0	IO	PWM
C2	PA1	IO	HZ	GPIO	SCL1_M	O	I2C1	INT1_S	O	HSTIF
B1	PA6	IO	HZ	GPIO	INT0_S	O	HSTIF	—	—	—
A4	VPP	—	—	TEST	—	—	—	—	—	—

*1 The used pin is determined by the HSTIF setting.

1.4.1 Pin Description

Power Supply

Pin name	I/O	Description	Polarity
GND	—	IO/core GND	—
VDD	—	IO power supply	—
VDDL	—	Core power supply (generated by the internal regulator)	—

SYSTEM

Pin name	I/O	Description	Polarity
CLK	I	Clock input (32.768 kHz)	—
BRMP	I	Remapping control input (for firmware update) Based on the BRMP pin setting at the time of the reset release, Bank0 is remapped. See “3.4.4 Boot/Remapping Function” for details.	—
RESET_N	I	Reset input	Negative

DEBUG Interface

Pin name	I/O	Description	Polarity
SWC	I	Serial clock of Serial Wire Debug Port	—
SWD	IO	Serial I/O data of Serial Wire Debug Port	—

Host Interface (HSTIF)

Pin name	I/O	Description	Polarity
SCL_S*2	I	SCL of I2C slave interface	—
SDA_S*2	IO	SDA of I2C slave interface	—
SCLK_S	I	SCLK of SPI slave interface	—
SDIO_S	IO	SDI and SDO of SPI slave interface (three-wire)	—
SDI_S	I	SDI of SPI slave interface (four-wire)	—
SCS_S	I	SCS of SPI slave interface	*1
SDO_S	O	SDO of SPI slave interface (four-wire)	—
INT0_S	O	Interrupt output 0 for host IF	Negative
INT1_S	O	Interrupt output 1 for host IF	Negative

*1 The polarity can be set by the software.
*2 3.6V tolerant in case of I^2C interface.

I2C master interface

Pin name	I/O	Description	Polarity
SDA0_M*	IO	SDA of I2C0 master interface	—
SCL0_M*	O	SCL of I2C0 master interface	—
SDA1_M*	IO	SDA of I2C1 master interface	—
SCL1_M*	O	SCL of I2C1 master interface	—

* 3.6V tolerant.

UART

Pin name	I/O	Description	Polarity
RXD0	I	UART received data	—
TXD0	O	UART transmitted data	—

PWM

Pin name	I/O	Description	Polarity
PWM0	IO	Output: PWM interface Input: PWM and timer clock input	—

GPIO

Pin name	I/O	Description	Polarity
PA0 to PA6	IO	GPIO (External interrupt function available)	—

TEST

Pin name	I/O	Description	Polarity
VPP	—	FLASH test pin	—

1.4.2 Handling of Unused Pins

Table 1-1 shows methods of terminating the unused pins.

Table 1-1 Termination of Unused Pins

Pin	Recommended pin handling
VPP	Open
BRMP	Open
SWC	Connect a pull-up resistor. (Recommended)
SWD	Connect a pull-up resistor. (Recommended)
SCL_S	Connect a pull-down resistor.
SDA_S	Connect a pull-down resistor.
PA0 to PA6	Open (Note)
SDA0_M, SCL0_M	Connect a pull-up resistor.

[Note]

It is recommended to set the unused input ports and input/output ports to the input mode with pull-down/pull-up resistor or the output mode since the supply current may become excessively large if the pins are left open in the high impedance input setting.

Chapter 2 CPU

2 CPU

2.1 Overview

A RISC processor manufactured by ARM.
It is a 32-bit processor for small size and low power consumption applications and has a 3-stage pipeline configuration. It implements the ARMv6-M architecture, and operates with 16-bit Thumb® instructions and Thumb®-2 instructions.
For details, see "Cortex®-M0 Technical Reference Manual","Cortex®-M0 Devices Generic User Guide ".

2.1.1 Features

- Multiplier can process 32 bit × 32 bit in one cycle (holding the lower 32 bits of the operation result).
- Serial wire debug port
- Little-Endian
- Built-in debug component with four break points and two watch points
- Wakeup from any interrupt is possible.
- WFI (Wait for Interrupts) supported
- WFE (Wait for Events) not supported
- SysTick not supported

2.2 Description of Registers

2.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0xE000_ED00	CPUID register	CPUID	R	32	0x410C_C200
0xE000_ED04	Interrupt control and state register	ICSR	R/W	32	0x0000_0000
0xE000_ED0C	Application interrupt and reset control register	AIRCR	R/W	32	0xFA05_0000
0xE000_ED10	System control register	SCR	R/W	32	0x0000_0000
0xE000_ED14	Configuration and control register	CCR	R	32	0x0000_0208
0xE000_ED1C	System handler priority register 2	SHPR2	R/W	32	0x0000_0000
0xE000_ED20	System handler priority register 3	SHPR3	R/W	32	0x0000_0000

For details of the register, see "Cortex®-M0 Devices Generic User Guide".

Chapter 3 Memory Space

3 Memory Space

3.1 Overview

In ML630Q791, various memories and registers are located in a 4 GB memory space, which is partitioned into 32 banks of 128 MB each.

3.2 Memory Map

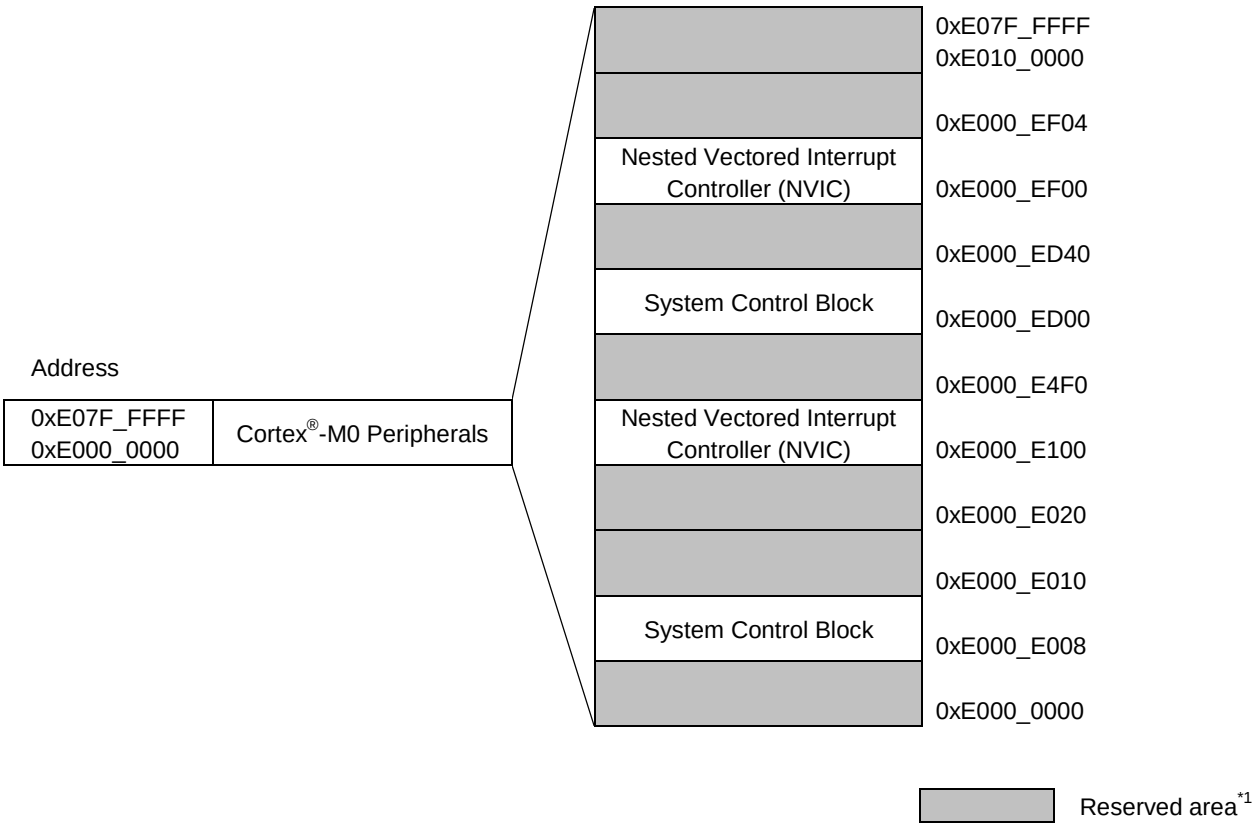
Figure 3-1 shows the memory map of ML630Q791.

Bank	4GB	Address	Memory space	
31		0xF800_0000		
30		0xF000_0000		
29		0xE800_0000		
28		0xE000_0000	Cortex®-M0 Peripherals	See Figure 3-1 (2)
27		0xD800_0000		
26		0xD000_0000		
25		0xC800_0000		
24	3GB	0xC000_0000		
23		0xB800_0000		
22		0xB000_0000		
21		0xA800_0000		
20		0xA000_0000		
19		0x9800_0000		
18		0x9000_0000		
17		0x8800_0000		
16	2GB	0x8000_0000		
15		0x7800_0000		
14		0x7000_0000		
13		0x6800_0000		
12		0x6000_0000		
11		0x5800_0000		
10		0x5000_0000		
9		0x4800_0000		
8	1GB	0x4000_0000	APB I/O	See Figure 3-1 (3)
7		0x3800_0000		
6		0x3000_0000		
5		0x2800_0000		
4		0x2000_0000	Work RAM (16 KB)	
3		0x1800_0000		
2		0x1000_0000	Internal Flash ROM (128 KB)	
1		0x0800_0000		
0	0GB	0x0000_0000	Remappable space	

: Reserved area[*1]

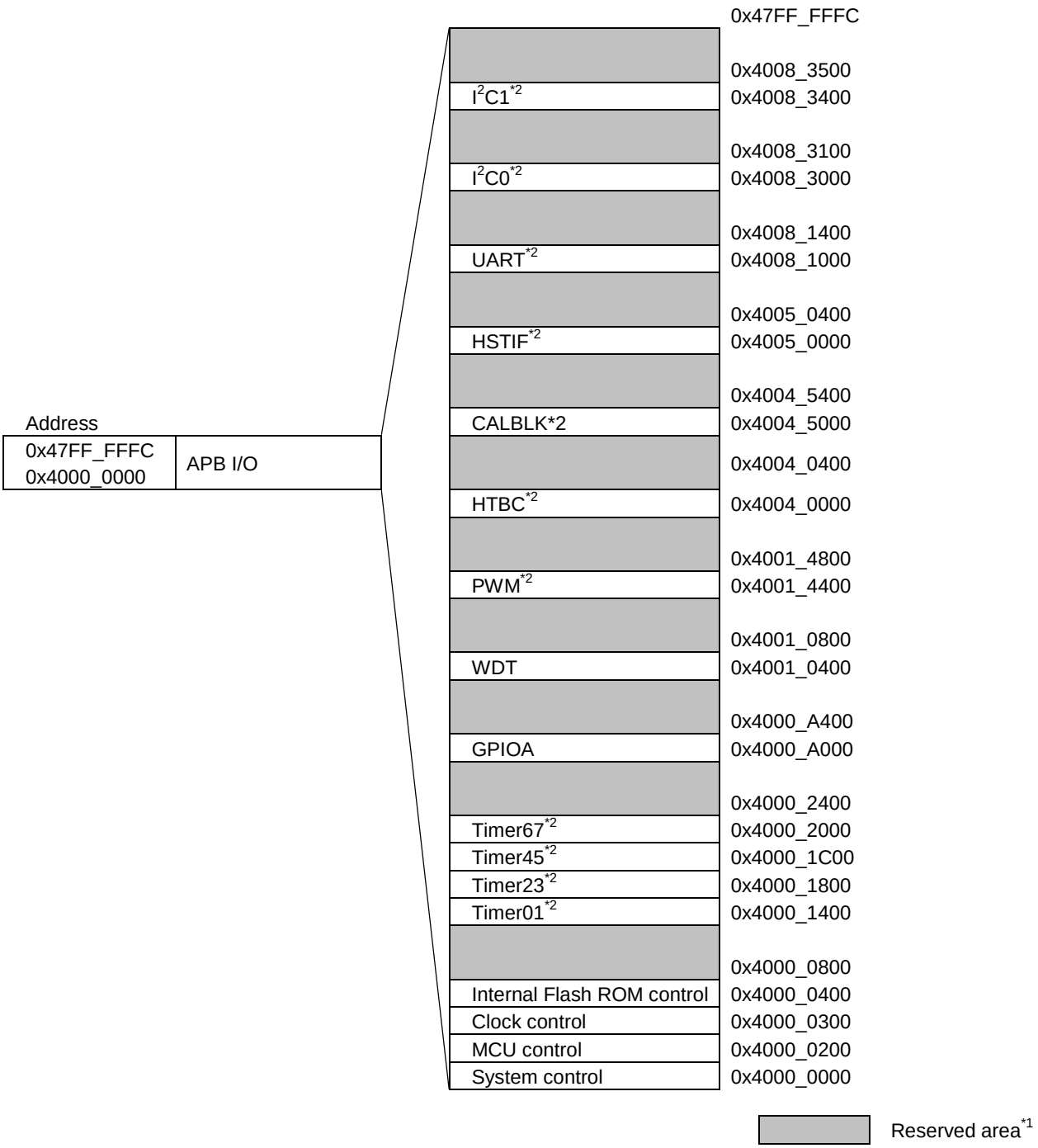
*1:Accessing any of reserved areas is prohibited. Proper operation cannot be guaranteed if accessed.

Figure 3-1 (1) ML630Q791 Memory Map



*1: Accessing any of reserved areas is prohibited. Proper operation cannot be guaranteed if accessed.

Figure 3-1 (2) Cortex®-M0 Peripherals Area Memory Map



*1:Accessing any of reserved areas is prohibited. Proper operation cannot be guaranteed if accessed.
*2:During individual block stop state, write data is ignored and read data is undefined. For details of individual block stop state, see Chapter 5.

Figure 3-1 (3) APB I/O Area Memory Map

3.3 Internal Memory

3.3.1 Internal Flash ROM

Table 3-1 shows the address range of the Flash ROM of Bank2.

Table 3-1 Address Range of Bank2 Memory

	Address range
Internal Flash ROM (128 KB)	0x1000_0000 to 0x1001_FFFF *1

*1:Accessing an address out of this address range in the same bank is prohibited. Proper operation cannot be guaranteed if accessed.
Address Range of 0x1001FE00 to 0x1001FFFF are the test data area. Erase/Program operation is prohibited.

The address holds the 28-bit product ID and 4-bit revision.

Address: 0x1001_FFFC
Access: R
Access size: 32 Bits
Initial value: 0x0630_7910

[Description of Bits]

- Bit [3:0]
  Represent the revision of this LSI.

- Bit [31:4]
  There bits represent the product ID of this LSI.
  ML630Q791: 0x0630791

3.3.2 Work RAM

Table 3-2 shows the address range of the work RAM of Bank4.

Table 3-2 Bank4 Address Range

	Address range
Work RAM (16 KB)	0x2000_0000 to 0x2000_3FFF *1

*1: Accessing an address out of this address range in the same bank is prohibited. Proper operation cannot be guaranteed if accessed.

3.4 Memory Controller Function

3.4.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4000_0010	Remapping control register	SYSCON_REMAP_CON	R/W	32	0x0000_0000
0x4000_0014	Remapping base address register	SYSCON_REMAP_BASE	R/W	32	0x1001_F000

3.4.2 Remapping Control Register (SYSCON_REMAP_CON)

Address: 0x4000_0010
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	REMAP_EN	REMAP[3:0]			
Access	–	–	–	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write a "0" for write.

[Description of Register]
This is a special function register (SFR) used to set remapping.

[Description of Each Bit]
REMAP[3:0] (bits 0-3)
REMAP_EN (bit 4)

REMAP_EN	REMAP [3:0]	Description
0	xxxx	Based on the BRMP pin that is set after the reset release, the following area is remapped to Bank0. H: The firmware update area of Flash ROM is remapped to Bank0. L: The area of address 0 of Flash ROM is remapped to Bank0.
1	0000	The internal Flash ROM is remapped to Bank0.
1	xxx1	The internal RAM is remapped to Bank0.
1	x100	The area starting at the address set by the REMAP_BASE register is remapped to Bank0.
1	Other than above	Setting prohibited

(x: Don't care)

[Notes on Setting]

1. Operation cannot be guaranteed if remapping is performed when the remapping processing program (instruction to set the remapping control register) is placed in Bank0. Be sure to place the remapping processing program in a Bank other than Bank0 when performing the remapping.

2. Remapping of Bank0 is performed as soon as this register is set.

3.4.3 Remapping Base Address Register (SYSCON_REMAP_BASE)

Address: 0x4000_0014
Access: R/W
Access size: 32 Bits
Initial value: 0x1001_F000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	REMAP_BASE[29:16]													
Access	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	REMAP_BASE[15:12]				–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	R/W	R/W	R/W	R/W	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
This is a special function register (SFR) used to set the base address of remapping. It is enabled only when REMAP_EN = 1 and REMAP[2:0] = ”100” are set in the remapping control register.

[Description of Each Bit]

REMAP_BASE[29:12]

The 4-KB area starting at the address set by this register is assigned starting from address 0.
The starting address must be an address in the memory space.
Flash ROM: 0x1000_0000-0x1001_FFFF
Work RAM: 0x2000_0000 - 0x2000_3FFF

* Area assigned to the remapping area is only 4 KB at the remapping by the remapping base address. If the space exceeding 4 KB is needed for ISP, for example, the program needs to jump to the area where the entity such as Flash ROM is placed (0x10000000 and after for Flash ROM) to be executed, except for the minimum necessary codes such as exception vectors.

3.4.4 Boot/Remapping Function

It is possible to allocate the following devices to Bank0 after booting by setting REMAP[3:0] of the remapping control register. Table 3-3 shows the allocations of Bank0 during booting and remapping.
Operation cannot be guaranteed if remapping is performed when the remapping processing program (instruction to set the remapping control register) is placed in Bank0. Be sure to place the remapping processing program in a Bank other than Bank0 when performing the remapping.

Table 3-3 Bank0 Allocations during Booting and Remapping

Boot/ Remapping	BRMP	REMAP _EN	REMAP [3:0]	Device	Remarks
Boot	0	0	xxxx	Internal Flash ROM	Start address: 0x1000_0000
	1			Internal Flash ROM	Start address: 0x1001_F000
Remapping	x	0	xxxx	Not remapped	
		1	0000	Internal Flash ROM	
			xxx1	Work RAM	
			x100	Work RAM/ Internal Flash ROM	The device placed at the address set by the remapping base address responds.
			Other than above	Setting prohibited	

x: Don't care the data.

3.5 Access Response for Memory Space

• An access made to a bank that has been set as not allocated returns an error response (*). For specifications of the space in a bank exceeding the allocated memory size, see Section 3.3.

*: Operation at error response
• If an error response is returned for access from CPU, a hard fault exception is generated.

Chapter 4 Reset Functions

4. Reset Functions

4.1 Overview

This LSI has the two reset functions shown below. If either of them is generated, this LSI enters the system reset mode.

- Reset by the RESET_N pin (external reset)
- Reset by SYSRESETREQ of Cortex®-M0 (software reset)

4.1.1 Features

- The RESET_N pin has an internal pull-up resistor

4.1.2 Configuration

Figure 4-1 shows the configuration of the reset generation circuit.

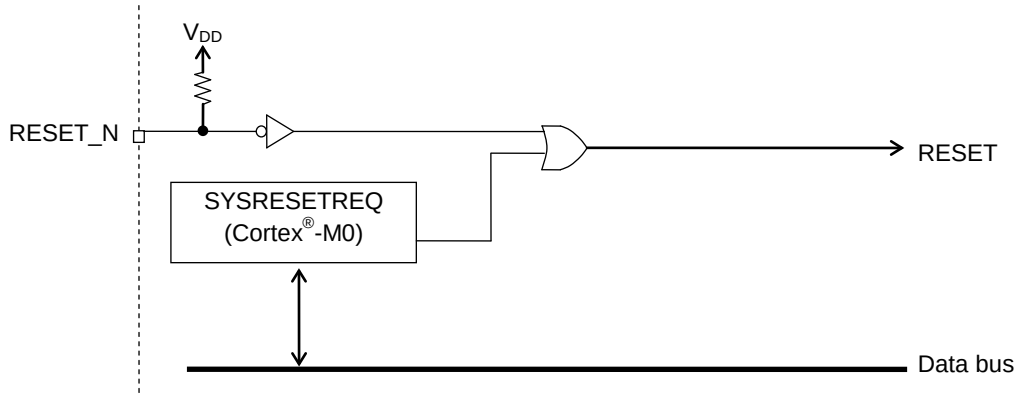


Figure 4-1 Configuration of Reset Generation Circuit

4.1.3 List of Pins

Pin name	I/O	Description
RESET_N	I	Reset input pin

4.2. Description of Registers

SYSRESETREQ is controlled by the SYSRESETREQ bit of the AIRCR register of Cortex®-M0. For details, see "Cortex-M0 Devices Generic User Guide".

4.3 Description of Operation

4.3.1 System Reset Mode

System reset has the highest priority among all the processing and any other processing being executed up to then is cancelled.
The system reset mode is set by any of the following causes.

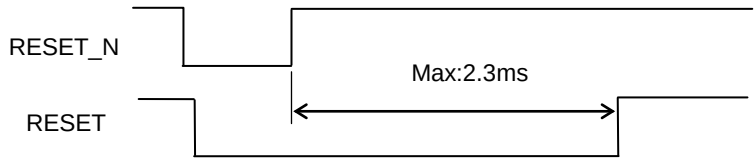
- Reset by the RESET_N pin (external reset)
- Reset by SYSRESETREQ of Cortex®-M0 (software reset)

In the system reset mode, the following processing is performed.

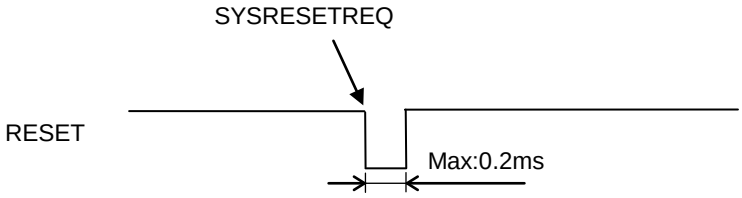
(1) The internal regulator is initialized. However, it is not initialized by a software reset.
(2) All the special function registers (SFRs) whose initial value is not undefined are initialized. See Appendix A "Registers" for the initial values of the SFRs. However, SYSCON_REMAP_CON and SYSCON_REMAP_BASE are not initialized by a software reset.
(3) CPU is initialized.
- All the registers in CPU are initialized.
- The program fetches the reset exception vector.

[Note]
In system reset mode, the contents of data memory and those of any SFR whose initial value is undefined are not initialized and are undefined. Initialize this area by the software.
The timing of release of reset differs between the external reset by the RESET_N pin and the software reset by SYSRESETREQ. The timing diagrams are shown below.

External reset by the RESET_N pin



Software reset by SYSRESETREQ



Chapter 5 MCU Control Function

5. MCU Control Function

5.1. Overview

This LSI includes power management, pin switching control, and other functions.

Operation of this LSI is categorized into the following three statuses:

(1) System reset mode
(2) Program run mode
(3) Sleep mode

This LSI can operate with a lower current consumption by powering down the unused function blocks (reset registers and stop clock supplies).

5.1.1 Features

- Retains the revision of this LSI.
- Uses block control function to power down the circuits of unused function blocks (reset registers and stop clock supplies).
- Controls switching of pin functions (switching to the secondary or tertiary function).
- Supports the sleep mode by the WFI instruction of Cortex®-M0.
  -Supports the low power mode selection by using the SLEEPDEEP bit.

  For details of WFI instruction and SLEEPDEEP bit, see "Cortex®-M0 Devices Generic User Guide".

5.2 Description of Registers

5.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4000_0200	Revision register	IDR	R	32	0x0630_7900
0x4000_0220	Peripheral clock enable register	PECLKEN	R/W	32	0x8000_0000
0x4000_0224	Peripheral clock disable register	PECLKDIS	R/W	32	0x0113_11FF
0x4000_0228	Peripheral reset enable register	PERSTEN	R/W	32	0x0113_11FF
0x4000_022C	Peripheral reset disable register	PERSTDIS	R/W	32	0x8000_0000
0x4000_0260	Port A mode setting register	PAMOD	R/W	32	0x0000_0000

5.2.2 Revision Register (IDR)

Address: 0x4000_0200
Access: R
Access size: 32 Bits
Initial value: 0x0630_7900

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	PID[27:12]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	1	1	0	0	0	1	1	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	PID[11:0]												PRV[3:0]			
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0

[Description of Register]
IDR holds the 28-bit product ID and 4-bit revision.

[Description of Bits]

- **PRV[3:0]** (bit 3-0)
  Represent the revision of this LSI.

- **PID[27:0]** (bit 31-4)
  PID0 to 27 represent the product ID of ML630Q791.
  It is necessary to refer to the Internal Flash ROM to identify ML630Q791.
  See Chapter 3, "Memory" for details.

5.2.3 Peripheral Clock Enable Register (PECLKEN)

Address: 0x4000_0220
Access: R/W
Access size: 32 Bits
Initial value: 0x8000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CE HST	-*	-*	-*	-*	-*	-*	CE CAL	-*	-*	-*	CE UART	-*	-*	CE I2C1	CE I2C0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	CE HTC	-*	-*	-*	CE PWM	CE TM7	CE TM6	CE TM5	CE TM4	CE TM3	CE TM2	CE TM1	CE TM0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

PECLKEN is a special function register (SFR) used to start the clock of each peripheral, and can be accessed only by writing "1". (Access by writing "0" is invalid.) When the peripheral clock disable register (PECLKDIS) is accessed by writing "1", the corresponding bit of PECLKEN is automatically reset to "0".
The clock enable state of each peripheral ("1" indicates the enable state) can be read when reading.

Description of Bits

Symbol name	Description	Related Bit		
		PERSTEN	PECLKDIS	PERSTDIS
CETMn (n=0-7)	To start the operation of Timer n (n = 0 to 7), write "1" to this bit.	RETMn (n=0-7)	CDTMn (n=0-7)	RDTMn (n=0-7)
CEPWM	To start the operation of PWM, write "1" to this bit.	REPWM	CDPWM	RDPWM
CEHTC	To start the operation of high-speed clock time base counter, write "1" to this bit.	REHTC	CDHTC	RDHTC
CEI2C0	To start the operation of I2C bus interface 0, write "1" to this bit.	REI2C0	CDI2C0	RDI2C0
CEI2C1	To start the operation of I2C bus interface 1, write "1" to this bit.	REI2C1	CDI2C1	RDI2C1
CEUART	To start the operation of UART, write "1" to this bit.	REUART	CDUART	RDUART
CECAL	To start the operation of arithmetic circuit, write "1" to this bit.	RECAL	CDCAL	RDCAL
CEHST	To start the operation of host interface, write "1" to this bit.	REHST	CDHST	RDHST

5.2.4 Peripheral Clock Disable Register (PECLKDIS)

Address: 0x4000_0224
Access: R/W
Access size: 32 Bits
Initial value: 0x0113_11FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CD HST	-*	-*	-*	-*	-*	-*	CD CAL	-*	-*	-*	CD UART	-*	-*	CD I2C1	CD I2C0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	CD HTC	-*	-*	-*	CD PWM	CD TM7	CD TM6	CD TM5	CD TM4	CD TM3	CD TM2	CD TM1	CD TM0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

PECLKDIS is a special function register (SFR) used to stop the clock of each peripheral, and can be accessed only by writing "1". (Access by writing "0" is invalid.) When the peripheral clock enable register (PECLKEN) is accessed by writing "1", the corresponding bit of PECLKDIS is automatically reset to "0".
The clock disable state of each peripheral ("1" indicates the disable state) can be read when reading.

Description of Bits

Symbol name	Description	Related Bit		
		PECLKEN	PERSTEN	PERSTDIS
CDTMn (n=0-7)	To stop the operation of Timer n (n = 0 to 7), write "1" to this bit.	CETMn (n=0-7)	RETMn (n=0-7)	RDTMn (n=0-7)
CDPWM	To stop the operation of PWM, write "1" to this bit.	CEPWM	REPWM	RDPWM
CDHTC	To stop the operation of high-speed clock time base counter, write "1" to this bit.	CEHTC	REHTC	RDHTC
CDI2C0	To stop the operation of I2C bus interface 0, write "1" to this bit.	CEI2C0	REI2C0	RDI2C0
CDI2C1	To stop the operation of I2C bus interface 1, write "1" to this bit.	CEI2C1	REI2C1	RDI2C1
CDUART	To stop the operation of UART, write "1" to this bit.	CEUART	REUART	RDUART
CDCAL	To stop the operation of arithmetic circuit, write "1" to this bit.	CECAL	RECAL	RDCAL
CDHST	To stop the operation of host interface, write "1" to this bit.	CEHST	REHST	RDHST

5.2.5 Peripheral Reset Enable Register (PERSTEN)

Address: 0x4000_0228
Access: R/W
Access size: 32 Bits
Initial value: 0x0113_11FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	RE HST	-*	-*	-*	-*	-*	-*	RE CAL	-*	-*	-*	RE UART	-*	-*	RE I2C1	RE I2C0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	RE HTC	-*	-*	-*	RE PWM	RE TM7	RE TM6	RE TM5	RE TM4	RE TM3	RE TM2	RE TM1	RE TM0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

PERSTEN is a special function register (SFR) used to assert the reset of each peripheral, and can be accessed only by writing "1". (Access by writing "0" is invalid.) The block can be stopped by setting the peripheral clock disable register (PECLKDIS) after setting PERSTEN. When the peripheral reset disable register (PERSTDIS) is accessed by writing "1", the corresponding bit of PERSTEN is automatically reset to "0".
The reset asserted state of each peripheral ("1" indicates the asserted state) can be read when reading.

Description of Bits

Symbol name	Description	Related Bit		
		PECLKEN	PECLKDIS	PERSTDIS
RETMn (n=0-7)	To stop the operation of Timer n (n = 0 to 7), write "1" to this bit.	CETMn (n=0-7)	CDTMn (n=0-7)	RDTMn (n=0-7)
REPWM	To stop the operation of PWM, write "1" to this bit.	CEPWM	CDPWM	RDPWM
REHTC	To stop the operation of high-speed clock time base counter, write "1" to this bit.	CEHTC	CDHTC	RDHTC
REI2C0	To stop the operation of I2C bus interface 0, write "1" to this bit.	CEI2C0	CDI2C0	RDI2C0
REI2C1	To stop the operation of I2C bus interface 1, write "1" to this bit.	CEI2C1	CDI2C1	RDI2C1
REUART	To stop the operation of UART, write "1" to this bit.	CEUART	CDUART	RDUART
RECAL	To stop the operation of arithmetic circuit, write "1" to this bit.	CECAL	CDCAL	RDCAL
REHST	To stop the operation of host interface, write "1" to this bit.	CEHST	CDHST	RDHST

5.2.6 Peripheral Reset Disable Register (PERSTDIS)

Address: 0x4000_022C
Access: R/W
Access size: 32 Bits
Initial value: 0x8000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	RD HST	-*	-*	-*	-*	-*	-*	RD CAL	-*	-*	-*	RD UART	-*	-*	RD I2C1	RD I2C0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	RD HTC	-*	-*	-*	RD PWM	RD TM7	RD TM6	RD TM5	RD TM4	RD TM3	RD TM2	RD TM1	RD TM0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

PERSTDIS is a special function register (SFR) used to release the reset of each peripheral, and can be accessed only by writing "1". (Access by writing "0" is invalid.) Set the peripheral clock enable register (PECLKEN) before setting PERSTDIS to supply a clock to the block. When the peripheral reset enable register (PERSTEN) is accessed by writing "1", the corresponding bit of PERSTDIS is automatically reset to "0".
The reset released state of each peripheral ("1" indicates the released state) can be read when reading.

Description of Bits

Symbol name	Description	Related Bit		
		PECLKEN	PECLKDIS	PERSTEN
RDTMn (n=0-7)	To start the operation of Timer n (n = 0 to 7), write "1" to this bit.	CETMn (n=0-7)	CDTMn (n=0-7)	RETMn (n=0-7)
RDPWM	To start the operation of PWM, write "1" to this bit.	CEPWM	CDPWM	REPWM
RDHTC	To start the operation of high-speed clock time base counter, write "1" to this bit.	CEHTC	CDHTC	REHTC
RDI2C0	To start the operation of I2C bus interface 0, write "1" to this bit.	CEI2C0	CDI2C0	REI2C0
RDI2C1	To start the operation of I2C bus interface 1, write "1" to this bit.	CEI2C1	CDI2C1	REI2C1
RDUART	To start the operation of UART, write "1" to this bit.*1	CEUART	CDUART	REUART
RDCAL	To start the operation of arithmetic circuit, write "1" to this bit.	CECAL	CDCAL	RECAL
RDHST	To start the operation of host interface, write "1" to this bit.	CEHST	CDHST	REHST

*1: To release individual UART stop, set CEUART =1 and then execute "NOP" twice or more to set RDUART = 1.

5.2.7 Port A Mode Setting Register (PAMOD)

Address: 0x4000_0260
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	PA6[1:0]		-*	-*	PA5[1:0]		-*	-*	PA4[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	PA3[1:0]		-*	-*	PA2[1:0]		-*	-*	PA1[1:0]		-*	-*	PA0[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

PAMOD is a special function register (SFR) used to select the primary/secondary/tertiary function of port.

Description of Bits

- **PA0** (bit 1, 0)
  PA0 is used to select the primary/secondary/tertiary function of Port PA0.

PA0[1:0]	Description
00	General-purpose input/output pin
01	I2C1 SDA pin
10	PWM PWM pin
11	Prohibited

- **PA1** (bit 5, 4)
  PA1 is used to select the primary/secondary/tertiary function of Port PA1.

PA1[1:0]	Description
00	General-purpose input/output pin
01	I2C1 SCL pin
10	Host interface INT1_S pin
11	Prohibited

- **PA2** (bit 9, 8)
  PA2 is used to select the primary/secondary function of Port PA2.

PA2[1:0]	Description
00	General-purpose input/output pin
01	UART RXD0 pin
10	Prohibited
11	Prohibited

- **PA3** (bit 13, 12)
  PA3 is used to select the primary/secondary function of Port PA3.

PA3[1:0]	Description
00	General-purpose input/output pin
01	UART TXD0 pin
10	Prohibited
11	Prohibited

- **PA4** (bit 17, 16)
  PA4 is used to select the primary/secondary function of Port PA4.

PA4[1:0]	Description
00	General-purpose input/output pin
01	Host interface SCS pin
10	Prohibited
11	Prohibited

- **PA5** (bit 21, 20)
  PA5 is used to select the primary/secondary function of Port PA5.

PA5[1:0]	Description
00	General-purpose input/output pin
01	Host interface SDO_S pin
10	Prohibited
11	Prohibited

- **PA6** (bit 25, 24)
  PA6 is used to select the primary/secondary function of Port PA6.

PA6[1:0]	Description
00	General-purpose input/output pin
01	Host interface INT0_S pin
10	Prohibited
11	Prohibited

5.3 Description of Operation

5.3.1 State Transition

Figure 5-1 shows an operating state transition diagram.

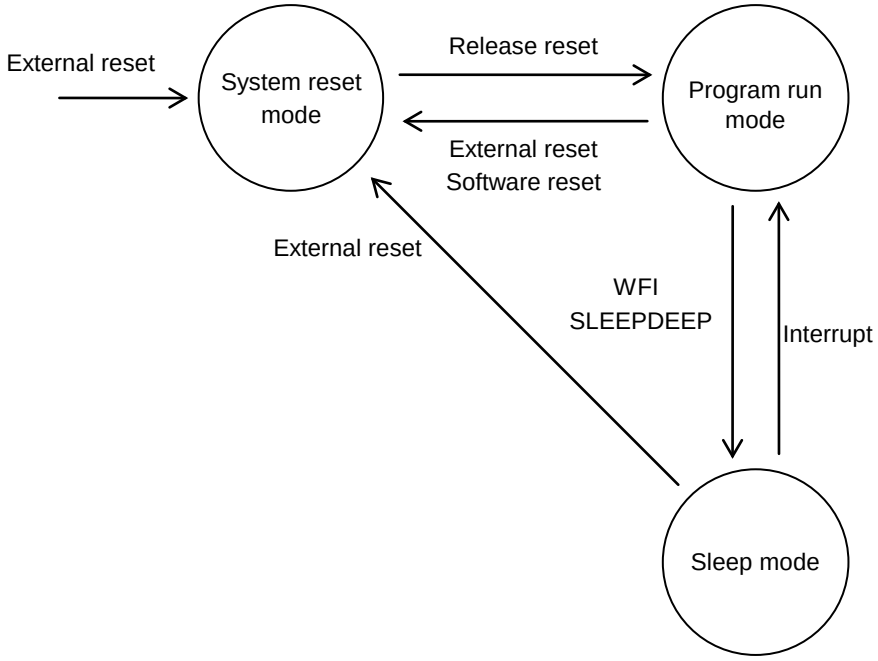


Figure 5-1 Operating State Transition Diagram

Operation mode	Function
System reset mode	The operating state goes into the system reset mode by an external reset (RESET_N pin) or software reset (SYSRESETREQ). The internal regulator, special function register (SFR), and CPU are initialized. For details, see Section 4.3.1 "System Reset Mode".
Program run mode	CPU can operate in this mode. The FLL control and the function stop control of each block are possible.
Sleep mode (SLEEPDEEP bit = 0)	Cortex®-M0 sleep mode. During the sleep mode, the settings of FLL and individual block that are set before entering the sleep mode are enabled. In this mode, the communication of I2C master/UART/host interface and the operation of arithmetic circuit/timer/PWM are possible.
Sleep mode (SLEEPDEEP bit = 1)	Cortex®-M0 sleep mode. During the sleep mode, the setting of individual block that is set before entering the sleep mode is enabled. FLL is automatically stopped when entering the sleep mode. Then, FLL is started or goes into the stop state according to the setting of the ENOSC bit of the FCON1 register when returning from the sleep mode due to an interrupt. In this mode, the communication of host interface and the operation of timer/PWM based on the low-speed clock are possible.

5.3.2 State of Each Operation Mode

The operation states of CPU, FLL, and individual block in each operation mode are as follows.

(○: Operates, ×: Stopped, △: Follows the register setting)

	Operation mode				
	After release of the system reset mode (External reset)	After release of the system reset mode (Software reset)	Program run mode	Sleep mode (SLEEPDEEP=0)	Sleep mode (SLEEPDEEP=1)
CPU	○	○	○	×	×
FLL	○(*1)	○(*2)	△	△	×
Internal regulator	○(*3)	○(*4)	○	○	○
Timer0	×	×	△	△(*5)	△(*6)
Timer1	×	×	△	△(*5)	△(*6)
Timer2	×	×	△	△(*5)	△(*6)
Timer3	×	×	△	△(*5)	△(*6)
Timer4	×	×	△	△(*5)	△(*6)
Timer5	×	×	△	△(*5)	△(*6)
Timer6	×	×	△	△(*5)	△(*6)
Timer7	×	×	△	△(*5)	△(*6)
PWM	×	×	△	△(*5)	△(*7)
HTBC	×	×	△	△(*5)	×
I2C0	×	×	△	△(*5)	×
I2C1	×	×	△	△(*5)	×
UART	×	×	△	△(*5)	×
Arithmetic circuit	×	×	△	△(*5)	△(*5)
Host interface	○	○	△	△(*5)	△(*5)
WDT	○	○	○	○	○
Other blocks	○	○	○	○	○

(*1) FLL is restarted.
(*2) FLL is not restarted (the lock state is maintained), but the system clock is switched to the low-speed clock.
(*3) The internal regulator is restarted by an external reset.
(*4) The internal regulator is not restarted by a software reset.
(*5) It follows the PECLKEN, PECLKDIS, PERSTEN, and PERSTDIS register settings made during the program run mode.
(*6) It follows the PECLKEN, PECLKDIS, PERSTEN, and PERSTDIS register settings made during the program run mode. It stops when HTBCLK is selected for the timer clock.
(*7) It follows the PECLKEN, PECLKDIS, PERSTEN, and PERSTDIS register settings made during the program run mode. It stops when HTBCLK is selected for the PWM clock.

Chapter 6 Clock

6. Clock

6.1 Overview

The clock of this LSI is supplied from the clock generation circuit and the time base counter.
Clock generation circuit generates the low-speed clock (LSCLK), high-speed clock (HSCLK), and system clock (SYSCLK) based on the external clock input, and supplies them. The low-speed clock and high-speed clock are used as clocks for the peripheral circuit and the time base counter, while the system clock is used as a clock for the CPU and memory.
As the time base counters, a low-speed time base counter (LTBC) and a high-speed time base counter (HTBC) are included. Low-speed time base counter supplies the divided clock by dividing the low-speed clock. High-speed time base counter supplies the divided clock by dividing the high-speed clock.

6.1.1 Features

- Low-speed clock supplies 32.768 kHz supplied from the external.
- High-speed clock supplies a clock generated by the internal FLL (Frequency Locked Loop).
- Low-speed time base counter generates the divided clock of 2.048 kHz or 256 Hz by dividing the low-speed clock.
- High-speed time base counter generates the divided clock HTBCLK by dividing the clock of 4 MHz. Dividing ratio can be changed.

6.1.2 Configuration

Figure 6-1 shows the configuration of the clock circuit.

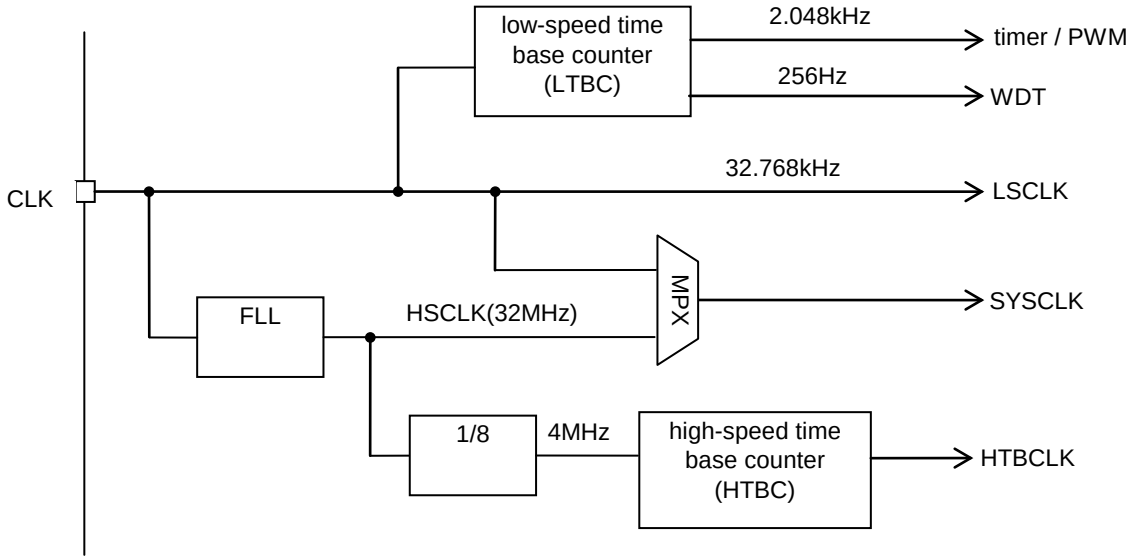


Figure 6-1 Configuration of Clock Circuit

6.2 Description of Registers

6.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4000_0300	Frequency control register 0	FCON0	R/W	32	0x0000_0000
0x4000_0304	Frequency control register 1	FCON1	R/W	32	0x0000_0002
0x4004_0000	High-speed time base counter frequency divide register	HTBDR	R/W	32	0x0000_0000

6.2.2 Frequency Control Register 0 (FCON0)

Address: 0x4000_0300
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	CLKDIV[3:0]			
Access	-	-	-	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

FCON0 is a special function register (SFR) used to control the high-speed clock generation circuit.

Description of Bits

- **CLKDIV[3:0]** (bit 3-0)
  This bit selects the frequency of high-speed clock.

CLKDIV[3:0]	Description
0	32 MHz CLK output (initial value)
Other than above	Setting prohibited

6.2.3 Frequency Control Register 1 (FCON1)

Address: 0x4000_0304
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	LFLL	-*	-*	-*	-*	-*	ENOSC	SYSCLK
Access	-	-	-	-	-	-	-	-	R	-	-	-	-	-	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	*1	0	0	0	0	0	1	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

*1 Depends on the state of FLL.

FCON1 is a special function register (SFR) to control the high-speed clock generation circuit and to select system clock.

Description of Bits

- **SYSCLK** (bit 0)
  The SYSCLK bit is used to select system clock. Either low-speed clock or high-speed clock can be selected. Switch the system clock to high-speed clock after LFLL bit indicates that FLL oscillation clock is available. Do not switch simultaneously the system clock to low-speed clock with stop of FLL oscillation by ENOSC bit.

SYSCLK	Description
0	Low-speed clock: LSCLK (initial value)
1	High-speed clock: HSCLK

- **ENOSC** (bit 1)
  The ENOSC bit is used to select enable/disable of the oscillator circuit of the internal FLL. Use the LFLL bit to check that the FLL oscillation clock is available before using the FLL oscillation clock.

ENOSC	Description
0	Stops internal FLL oscillation
1	Enables internal FLL oscillation (initial value)

- **LFLL** (bit 7)
  The LFLL bit is used as a flag to indicate the oscillation state of the internal FLL.
  When LFLL is "1", it indicates that the FLL oscillation clock is available. When LFLL is "0", it indicates that FLL is inactive or the frequency of the FLL oscillation clock is not within the specification range.
  LFLL is a read-only bit.

LFLL	Description
0	FLL oscillation clock is disabled (initial value)
1	FLL oscillation clock is enabled

6.2.4 High-Speed Time Base Counter Frequency Divide Register (HTBDR)

Address: 0x4004_0000
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	HTD[3:0]			
Access	-	-	-	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

HTBDR is a special function register (SFR) to set the dividing ratio of the 4-bit, 1/n counter.

Description of Bits

- HTD[3:0] (bit 3-0)

The HTD[3:0] bits are used to set the dividing ratio of the 4-bit, 1/n counter. The frequency divide ratios selectable include 1/1 to 1/16.
The 4 MHz clock is divided to output a clock.

HTD[3:0]				Description
				Dividing ratio
0	0	0	0	1/16 (Initial value)
0	0	0	1	1/15
0	0	1	0	1/14
0	0	1	1	1/13
0	1	0	0	1/12
0	1	0	1	1/11
0	1	1	0	1/10
0	1	1	1	1/9
1	0	0	0	1/8
1	0	0	1	1/7
1	0	1	0	1/6
1	0	1	1	1/5
1	1	0	0	1/4
1	1	0	1	1/3
1	1	1	0	1/2
1	1	1	1	1/1

6.3 Description of Operation

6.3.1 Low-Speed Clock

For the low-speed clock, an external input clock is used. The clock of 32.768 kHz should be input.

6.3.2 High-Speed Clock

For the high-speed clock (HSCLK), a clock of 32 MHz generated by the internal FLL is used. A clock of 4 MHz to which the high-speed clock divided is input to High-Speed Time Base Counter.

6.3.2.1 Internal FLL Oscillation

The internal FLL oscillation generates a clock of 31.998 MHz ± 5% to which the clock of 32.768 kHz (Low-Speed clock) is multiplied. When the frequency of high-speed oscillation clock reaches within 31.998 MHz ± 5%, the LFLL flag of FCON1 is set to "1".

6.3.3 Low-Speed Time Base Counter

Each of the low-speed time base counter (LTBC) outputs is used as an operation clock for peripheral circuits.

6.3.4 High-Speed Time Base Counter

The high-speed time base counter is configured as a 4-bit 1/n counter (n = 1 to 16).
In the 4-bit 1/n counter, the divided clock (1/16 x 4MHz to 1/1 x 4MHz) selected by the high-speed time base counter frequency divide register (HTBDR) is generated as HTBCLK. HTBCLK is used as the operation clock of timer and PWM.
Figure 6-2 shows the output waveform of HTBCLK.

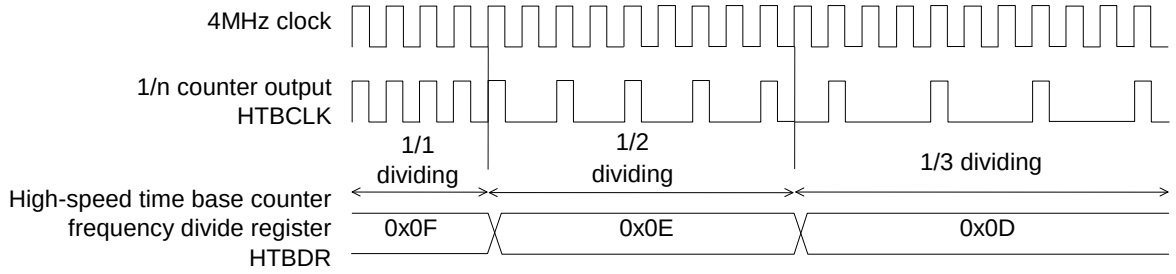


Figure 6-2 Output Waveform of HTBCLK

Chapter 7 Interrupt

7. Interrupt

7.1 Overview

This LSI has 22 interrupt sources (External interrupts: 7 sources, Internal interrupts: 15 sources) and a software interrupt (SVC).
For details of the interrupt function, see the section about NVIC of " Cortex®-M0 Devices Generic User Guide ".
For details of each interrupt, see the following chapters:

Chapter 8, "Timer"
Chapter 9, "PWM"
Chapter 10, "Watchdog Timer"
Chapter 11, "Host Interface"
Chapter 12, "I^2C Bus Interface"
Chapter 13, "UART"
Chapter 14 "Arithmetic Circuit"
Chapter 15, "GPIO"

7.1.1 Features

- Non-maskable interrupt source: 1 (Internal sources: 1)
- Maskable interrupt sources: 21 (Internal sources: 14, External sources: 7)
- Software interrupt (SVC)
- Four interrupt priority levels supported

7.2 Description of Registers

7.2.1 List of Registers

Address	Name	Symbol (Byte)	R/W	Size [bits]	Initial value
0xE000_E100	Interrupt set-enable register	NVIC_ISER	R/W	32	0x0000_0000
0xE000_E180	Interrupt clear-enable register	NVIC_ICER	R/W	32	0x0000_0000
0xE000_E200	Interrupt set-pending register	NVIC_ISPR	R/W	32	0x0000_0000
0xE000_E280	Interrupt clear-pending register	NVIC_ICPR	R/W	32	0x0000_0000
0xE000_E400	Interrupt priority register 0	NVIC_IPR0	R/W	32	0x0000_0000
0xE000_E404	Interrupt priority register 1	NVIC_IPR1	R/W	32	0x0000_0000
0xE000_E408	Interrupt priority register 2	NVIC_IPR2	R/W	32	0x0000_0000
0xE000_E40C	Interrupt priority register 3	NVIC_IPR3	R/W	32	0x0000_0000
0xE000_E410	Interrupt priority register 4	NVIC_IPR4	R/W	32	0x0000_0000
0xE000_E414	Interrupt priority register 5	NVIC_IPR5	R/W	32	0x0000_0000
0xE000_E418	Interrupt priority register 6	NVIC_IPR6	R/W	32	0x0000_0000
0xE000_E41C	Interrupt priority register 7	NVIC_IPR7	R/W	32	0x0000_0000

For details of the interrupt registers, see the section about NVIC of " Cortex-M0 User Guide ".

7.2.2 Correspondence with Bits

Table 7-1 shows the bit of each register for each interrupt number.

Table 7-1 Corresponding Bits

Interrupt number	NVIC_ISER	NVIC_ICER	NVIC_ISPR	NVIC_ICPR	NVIC_IPRn
NMI	-	-	-	-	-
IRQ[0]	SETENA[0]	CLRENA[0]	SETPEND[0]	CLRPEND[0]	NVIC_IPR0:PRI_0[1:0]
IRQ[1]	SETENA[1]	CLRENA[1]	SETPEND[1]	CLRPEND[1]	NVIC_IPR0:PRI_1[1:0]
IRQ[2]	SETENA[2]	CLRENA[2]	SETPEND[2]	CLRPEND[2]	NVIC_IPR0:PRI_2[1:0]
IRQ[3]	SETENA[3]	CLRENA[3]	SETPEND[3]	CLRPEND[3]	NVIC_IPR0:PRI_3[1:0]
IRQ[4]	SETENA[4]	CLRENA[4]	SETPEND[4]	CLRPEND[4]	NVIC_IPR1:PRI_4[1:0]
IRQ[5]	SETENA[5]	CLRENA[5]	SETPEND[5]	CLRPEND[5]	NVIC_IPR1:PRI_5[1:0]
IRQ[6]	SETENA[6]	CLRENA[6]	SETPEND[6]	CLRPEND[6]	NVIC_IPR1:PRI_6[1:0]
IRQ[7]	SETENA[7]	CLRENA[7]	SETPEND[7]	CLRPEND[7]	NVIC_IPR1:PRI_7[1:0]
IRQ[8]	SETENA[8]	CLRENA[8]	SETPEND[8]	CLRPEND[8]	NVIC_IPR2:PRI_8[1:0]
IRQ[9]	SETENA[9]	CLRENA[9]	SETPEND[9]	CLRPEND[9]	NVIC_IPR2:PRI_9[1:0]
IRQ[10]	SETENA[10]	CLRENA[10]	SETPEND[10]	CLRPEND[10]	NVIC_IPR2:PRI_10[1:0]
IRQ[11]	SETENA[11]	CLRENA[11]	SETPEND[11]	CLRPEND[11]	NVIC_IPR2:PRI_11[1:0]
IRQ[12]	SETENA[12]	CLRENA[12]	SETPEND[12]	CLRPEND[12]	NVIC_IPR3:PRI_12[1:0]
IRQ[13]	SETENA[13]	CLRENA[13]	SETPEND[13]	CLRPEND[13]	NVIC_IPR3:PRI_13[1:0]
IRQ[14]	SETENA[14]	CLRENA[14]	SETPEND[14]	CLRPEND[14]	NVIC_IPR3:PRI_14[1:0]
IRQ[15]	SETENA[15]	CLRENA[15]	SETPEND[15]	CLRPEND[15]	NVIC_IPR3:PRI_15[1:0]
IRQ[16]	SETENA[16]	CLRENA[16]	SETPEND[16]	CLRPEND[16]	NVIC_IPR4:PRI_16[1:0]
IRQ[17]	SETENA[17]	CLRENA[17]	SETPEND[17]	CLRPEND[17]	NVIC_IPR4:PRI_17[1:0]
IRQ[18]	SETENA[18]	CLRENA[18]	SETPEND[18]	CLRPEND[18]	NVIC_IPR4:PRI_18[1:0]
IRQ[19]	SETENA[19]	CLRENA[19]	SETPEND[19]	CLRPEND[19]	NVIC_IPR4:PRI_19[1:0]
IRQ[20]	SETENA[20]	CLRENA[20]	SETPEND[20]	CLRPEND[20]	NVIC_IPR5:PRI_20[1:0]
IRQ[21]	SETENA[21]	CLRENA[21]	SETPEND[21]	CLRPEND[21]	NVIC_IPR5:PRI_21[1:0]
IRQ[22]	SETENA[22]	CLRENA[22]	SETPEND[22]	CLRPEND[22]	NVIC_IPR5:PRI_22[1:0]
IRQ[23]	SETENA[23]	CLRENA[23]	SETPEND[23]	CLRPEND[23]	NVIC_IPR5:PRI_23[1:0]
IRQ[24]	SETENA[24]	CLRENA[24]	SETPEND[24]	CLRPEND[24]	NVIC_IPR6:PRI_24[1:0]
IRQ[25]	SETENA[25]	CLRENA[25]	SETPEND[25]	CLRPEND[25]	NVIC_IPR6:PRI_25[1:0]
IRQ[26]	SETENA[26]	CLRENA[26]	SETPEND[26]	CLRPEND[26]	NVIC_IPR6:PRI_26[1:0]
IRQ[27]	SETENA[27]	CLRENA[27]	SETPEND[27]	CLRPEND[27]	NVIC_IPR6:PRI_27[1:0]
IRQ[28]	SETENA[28]	CLRENA[28]	SETPEND[28]	CLRPEND[28]	NVIC_IPR7:PRI_28[1:0]
IRQ[29]	SETENA[29]	CLRENA[29]	SETPEND[29]	CLRPEND[29]	NVIC_IPR7:PRI_29[1:0]
IRQ[30]	SETENA[30]	CLRENA[30]	SETPEND[30]	CLRPEND[30]	NVIC_IPR7:PRI_30[1:0]
IRQ[31]	SETENA[31]	CLRENA[31]	SETPEND[31]	CLRPEND[31]	NVIC_IPR7:PRI_31[1:0]

7.3 Description of Operation

For 21 sources which do not include the watchdog timer interrupt (WDTINT), interrupt enable/disable is controlled by NVIC_ISER and NVIC_ICER. WDTINT is a non-maskable interrupt.
When the interrupt conditions are satisfied, the CPU reads the exception handler start address from the vector table address determined for each interrupt source and starts executing the exception handler.
Table 7-2 lists the interrupt sources.

Table 7-2 Interrupt Sources

Interrupt number	Interrupt source	Symbol	Vector table address
NMI	Watchdog timer interrupt	WDTINT	0x0000_0008
IRQ[0]	-	-	0x0000_0040
IRQ[1]	-	-	0x0000_0044
IRQ[2]	Port PA1 interrupt	PA1INT	0x0000_0048
IRQ[3]	Timer 0 interrupt	TM0INT	0x0000_004C
IRQ[4]	Timer 1 interrupt	TM1INT	0x0000_0050
IRQ[5]	PWM interrupt	PWMINT	0x0000_0054
IRQ[6]	Port PA0 interrupt	PA0INT	0x0000_0058
IRQ[7]	I^2C bus 0 interrupt	I2C0INT	0x0000_005C
IRQ[8]	-	-	0x0000_0060
IRQ[9]	Port PA4 interrupt	PA4INT	0x0000_0064
IRQ[10]	-	-	0x0000_0068
IRQ[11]	-	-	0x0000_006C
IRQ[12]	Timer 2 interrupt	TM2INT	0x0000_0070
IRQ[13]	Timer 3 interrupt	TM3INT	0x0000_0074
IRQ[14]	Arithmetic circuit interrupt	CALINT	0x0000_0078
IRQ[15]	UART interrupt	UAINT	0x0000_007C
IRQ[16]	-	-	0x0000_0080
IRQ[17]	-	-	0x0000_0084
IRQ[18]	Host IF interrupt	HSTINT	0x0000_0088
IRQ[19]	Port PA5 interrupt	PA5INT	0x0000_008C
IRQ[20]	I^2C bus 1 interrupt	I2C1INT	0x0000_0090
IRQ[21]	-	-	0x0000_0094
IRQ[22]	Port PA2 interrupt	PA2INT	0x0000_0098
IRQ[23]	Port PA3 interrupt	PA3INT	0x0000_009C
IRQ[24]	-	-	0x0000_00A0
IRQ[25]	-	-	0x0000_00A4
IRQ[26]	Timer 4 interrupt	TM4INT	0x0000_00A8
IRQ[27]	Timer 5 interrupt	TM5INT	0x0000_00AC
IRQ[28]	Port PA6 interrupt	PA6INT	0x0000_00B0
IRQ[29]	-	-	0x0000_00B4
IRQ[30]	Timer 6 interrupt	TM6INT	0x0000_00B8
IRQ[31]	Timer 7 interrupt	TM7INT	0x0000_00BC

[Note]

- When multiple interrupts are generated concurrently, they are processed starting from the highest priority level, and the lower-priority interrupts are pending. If they have the same priority level, the interrupt with a smaller interrupt number has higher priority.
  Please define vector tables for all unused interrupts for fail safe.

Chapter 8 Timer

8. Timer

8.1 Overview

This LSI includes 8 channels of 8-bit timers.
For input clocks, see Chapter 6, "Clock".

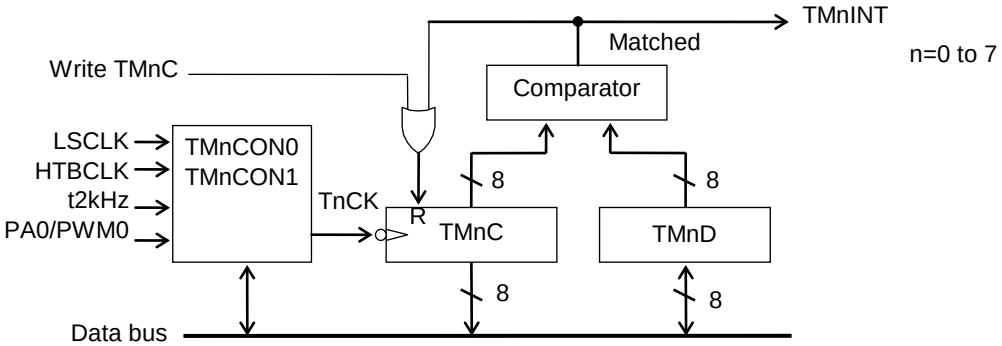
8.1.1 Features

- The timer interrupt (TMnINT) is generated when the values of timer counter register (TMnC, n=0 to 7) and timer data register (TMnD) coincide.
- Any combination of timer 0 and timer 1, timer 2 and timer 3, timer 4 and timer 5, or timer 6 and timer 7 can be used as a 16-bit timer.
- For the timer clock, the low-speed clock (LSCLK), high-speed time base clock (HTBCLK), low-speed time base clock t2kHz (2.048 kHz), or external clock (shared with PWM) can be selected.

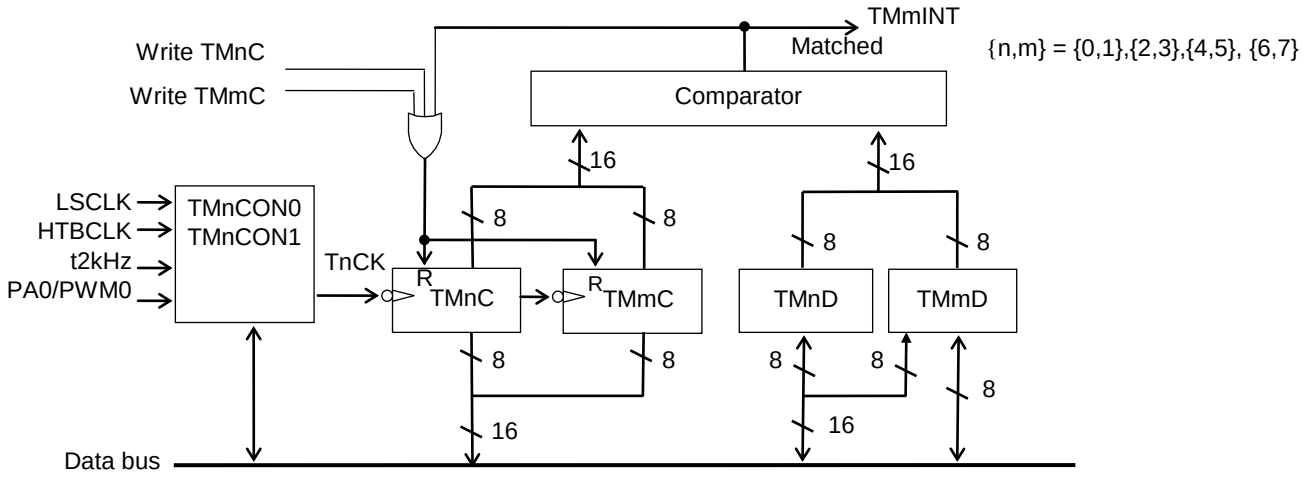
8.1.2 Configuration

Figure 8-1 shows the configuration of the timers.

TMnCON0 : Timer control register 0
TMnCON1 : Timer control register 1
TMmD, TMnD : Timer data register
TMmC, TMnC : Timer counter register



(a) In 8-bit Timer Mode (Timers 0 to 7)



(b) In 16-bit Timer Mode (Timers 0 to 7)

Figure 8-1 Timer Configuration

8.1.3 List of Pins

Pin name	I/O	Description
PA0	I	Timer external clock input pin Used for the tertiary function of the PA0 pin.

8.2 Description of Registers

8.2.1 List of Registers

ch	Address	Name	Symbol	R/W	Size [bits]	Initial value
0	0x4000_1400	Timer 0 data register	TM0D	R/W	32	0x0000_00FF
	0x4000_1404	Timer 0 counter register	TM0C	R/W	32	0x0000_0000
	0x4000_1408	Timer 0 control register 0	TM0CON0	R/W	32	0x0000_0000
	0x4000_140C	Timer 0 control register 1	TM0CON1	R/W	32	0x0000_0000
1	0x4000_1410	Timer 1 data register	TM1D	R/W	32	0x0000_00FF
	0x4000_1414	Timer 1 counter register	TM1C	R/W	32	0x0000_0000
	0x4000_1418	Timer 1 control register 0	TM1CON0	R/W	32	0x0000_0000
	0x4000_141C	Timer 1 control register 1	TM1CON1	R/W	32	0x0000_0000
2	0x4000_1800	Timer 2 data register	TM2D	R/W	32	0x0000_00FF
	0x4000_1804	Timer 2 counter register	TM2C	R/W	32	0x0000_0000
	0x4000_1808	Timer 2 control register 0	TM2CON0	R/W	32	0x0000_0000
	0x4000_180C	Timer 2 control register 1	TM2CON1	R/W	32	0x0000_0000
3	0x4000_1810	Timer 3 data register	TM3D	R/W	32	0x0000_00FF
	0x4000_1814	Timer 3 counter register	TM3C	R/W	32	0x0000_0000
	0x4000_1818	Timer 3 control register 0	TM3CON0	R/W	32	0x0000_0000
	0x4000_181C	Timer 3 control register 1	TM3CON1	R/W	32	0x0000_0000
4	0x4000_1C00	Timer 4 data register	TM4D	R/W	32	0x0000_00FF
	0x4000_1C04	Timer 4 counter register	TM4C	R/W	32	0x0000_0000
	0x4000_1C08	Timer 4 control register 0	TM4CON0	R/W	32	0x0000_0000
	0x4000_1C0C	Timer 4 control register 1	TM4CON1	R/W	32	0x0000_0000
5	0x4000_1C10	Timer 5 data register	TM5D	R/W	32	0x0000_00FF
	0x4000_1C14	Timer 5 counter register	TM5C	R/W	32	0x0000_0000
	0x4000_1C18	Timer 5 control register 0	TM5CON0	R/W	32	0x0000_0000
	0x4000_1C1C	Timer 5 control register 1	TM5CON1	R/W	32	0x0000_0000
6	0x4000_2000	Timer 6 data register	TM6D	R/W	32	0x0000_00FF
	0x4000_2004	Timer 6 counter register	TM6C	R/W	32	0x0000_0000
	0x4000_2008	Timer 6 control register 0	TM6CON0	R/W	32	0x0000_0000
	0x4000_200C	Timer 6 control register 1	TM6CON1	R/W	32	0x0000_0000
7	0x4000_2010	Timer 7 data register	TM7D	R/W	32	0x0000_00FF
	0x4000_2014	Timer 7 counter register	TM7C	R/W	32	0x0000_0000
	0x4000_2018	Timer 7 control register 0	TM7CON0	R/W	32	0x0000_0000
	0x4000_201C	Timer 7 control register 1	TM7CON1	R/W	32	0x0000_0000

8.2.2 Timer n Data Register (TMnD: n = 0, 2, 4, 6)

Address: 0x4000_1400 (TM0D)
0x4000_1800 (TM2D)
0x4000_1C00 (TM4D)
0x4000_2000 (TM6D)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_00FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	TnD[15:8]								TnD[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMnD is a special function register (SFR) to set the value to be compared with the timer n counter register (TMnC) value.
TnD[15:8] can be accessed only when the 16-bit timer mode is set. The initial values of TnD[15:8] are 0xFF in the 16-bit timer mode.

[Note]
Set TMnD when the timer n stops (When TnSTAT of TMnCON1 register is "0").
Writing "0x0000_0000" to TMnD works in the same way as "0x0000_0001".

8.2.3 Timer m Data Register (TMmD: m = 1, 3, 5, 7)

Address: 0x4000_1410 (TM1D)
0x4000_1810 (TM3D)
0x4000_1C10 TM5D)
0x4000_2010 (TM7D)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_00FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	TmD[7:0]							
Access	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMmD is a special function register (SFR) to set the value to be compared with the timer m counter register (TMmC) value.
The value set to TnD[15:8] (n = 0, 2, 4, 6) is reflected in the 16-bit timer mode.

[Note]
Set TMmD when the timer m stops (When TmSTAT of TMmCON1 register is "0").
Writing "0x0000_0000" to TMmD works in the same way as "0x0000_0001".

8.2.4 Timer n Counter Register (TMnC: n = 0, 2, 4, 6)

Address: 0x4000_1404 (TM0C)
0x4000_1804 (TM2C)
0x4000_1C04 (TM4C)
0x4000_2004 (TM6C)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	TnC[15:8]								TnC[7:0]							
Access	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMnC is a special function register (SFR) that functions as an 8/16-bit binary counter.
When a write operation is performed to TMnC, TMnC is set to "0x0000_0000". The data that is written is meaningless.
In the 16-bit timer mode, a write operation to either the TMnC or TM(n+1)C register sets "0x0000_0000" to both of them. (n=0,2,4,6)

During timer operation, the TMnC content may not be read depending on the conditions of the timer clock and the system clock.
Table 8-1 shows whether a TMnC read is enabled or disabled during timer operation for each condition of the timer clock and system clock.

Table 8-1 TMnC Read Enable/Disable during Timer Operation

Timer clock TnCK	System clock SYSCLK	TMnC read enable/disable
LSCLK	LSCLK	Read enabled.
	HSCLK	Read enabled. However, to prevent the reading of uncertain data during incremental counting, read TMnC twice and check that the results match.
HTBCLK	LSCLK	Read disabled.
	HSCLK	Read enabled.
t2kHz	LSCLK	Read enabled. However, to prevent the reading of uncertain data during incremental counting, read TMnC twice and check that the results match.
	HSCLK	Read enabled. However, to prevent the reading of uncertain data during incremental counting, read TMnC twice and check that the results match.
External clock	LSCLK	Read disabled.
	HSCLK	

8.2.5 Timer m Counter Register (TMmC: m = 1, 3, 5, 7)

Address: 0x4000_1414 (TM1C)
0x4000_1814 (TM3C)
0x4000_1C14 (TM5C)
0x4000_2014 (TM7C)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	TmC[7:0]							
Access	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMmC is a special function register (SFR) that functions as an 8-bit binary counter.
When a write operation is performed to TMmC, TMmC is set to "0x0000_0000". The data that is written is meaningless.
In the 16-bit timer mode, a write operation to either the TMnC or TM(n+1)C register sets "0x0000_0000" to both of them. (n=0,2,4,6)
In the 16-bit timer mode, the counter value is read from TnC[15:0] of the timer n counter register (n = 0, 2, 4, 6).

During timer operation, the TMmC content may not be read depending on the conditions of the timer clock and the system clock.
Table 8-1 shows whether a TMmC read is enabled or disabled during timer operation for each condition of the timer clock and system clock.

Table 8-2 TMmC Read Enable/Disable during Timer Operation

Timer clock TmCK	System clock SYSCLK	TMmC read enable/disable
LSCLK	LSCLK	Read enabled.
	HSCLK	Read enabled. However, to prevent the reading of undefined data during incremental counting, read TMmC twice and check that the results match.
HTBCLK	LSCLK	Read disabled.
	HSCLK	Read enabled.
t2kHz	LSCLK	Read enabled. However, to prevent the reading of undefined data during incremental counting, read TMmC twice and check that the results match.
	HSCLK	Read enabled. However, to prevent the reading of undefined data during incremental counting, read TMmC twice and check that the results match.
External clock	LSCLK	Read disabled.
	HSCLK	

8.2.6 Timer n Control Register 0 (TMnCON0: n = 0, 2, 4, 6)

Address: 0x4000_1408 (TM0CON0)
0x4000_1808 (TM2CON0)
0x4000_1C08 (TM4CON0)
0x4000_2008 (TM6CON0)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	TxxM16	TnCS[1:0]	
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMnCON0 is a special function register (SFR) used to control the Timer n.
Rewrite TMnCON0 while the timer n is stopped (TnSTAT of the TMnCON1 register is "0").

Description of Bits

- **TnCS[1:0]** (bit 1-0)
  TnCS[1:0] is used for selecting the operation clock of the timer n. LSCLK, HTBCLK, t2kHz, or the external clock can be selected by these bits.

TnCS[1: 0]		Description
0	0	LSCLK (initial value)
0	1	HTBCLK
1	0	t2kHz (2.048 kHz)
1	1	External clock (PA0/TIMER)

- **TxxM16** (bit 2)
  The TxxM16 bit is used for selecting the operation mode of timer n and timer n+1. (n=0,2,4,6 xx=01,23,45,67)
  In the 8-bit timer mode, each of timer n and timer n+1 operates independently as an 8-bit timer.
  In the 16-bit timer mode, timer n and timer n+1 are connected and they operate as a 16-bit timer.
  In the 16-bit timer mode, timer n+1 is incremented by a timer n overflow signal.
  A timer n interrupt (TMnINT) is not generated.

TxxM16	Description
0	8-bit timer mode (initial value)
1	16-bit timer mode

8.2.7 Timer m Control Register 0 (TMmCON0: m = 1, 3, 5, 7)

Address: 0x4000_1418 (TM1CON0)
0x4000_1818 (TM3CON0)
0x4000_1C18 (TM5CON0)
0x4000_2018 (TM7CON0)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	TmCS[1:0]	
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMmCON0 is a special function register (SFR) used to control the Timer m.
Rewrite TMmCON0 while the timer m is stopped (TmSTAT of the TMmCON1 register is "0").

Description of Bits

- **TmCS[1:0]** (bits 1-0)
  TmCS[1:0] is used for selecting the operation clock of the timer m. LSCLK, HTBCLK, t2kHz, or the external clock can be selected by these bits.
  When the 16-bit timer mode is selected, the value of this bit is invalid.

TmCS[1:0]		Description
0	0	LSCLK (initial value)
0	1	HTBCLK
1	0	t2kHz (2.048kHz)
1	1	External clock (PA0/TIMER)

8.2.8 Timer n Control Register 1 (TMnCON1: n = 0, 2, 4, 6)

Address: 0x4000_140C (TM0CON1)
0x4000_180C (TM2CON1)
0x4000_1C0C (TM4CON1)
0x4000_200C (TM6CON1)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	TnSTAT	–*	–*	–*	–*	–*	–*	TnRUN
Access	–	–	–	–	–	–	–	–	R	–	–	–	–	–	–	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMnCON1 is a special function register (SFR) used to control the Timer n.

Description of Bits

- **TnRUN** (bit 0)
  The TnRUN bit is used for controlling stop/start of timer n.

TnRUN	Description
0	Stops counting
1	Starts counting

- **TnSTAT** (bit 7)
  The TnSTAT bit is used for indicating "counting stopped"/"counting in progress" of timer n.

TnSTAT	Description
0	Counting stopped
1	Counting in progress

8.2.9 Timer m Control Register 1 (TMmCON1: m = 1, 3, 5, 7)

Address: 0x4000_141C (TM1CON1)
0x4000_181C (TM3CON1)
0x4000_1C1C (TM5CON1)
0x4000_201C TM7CON1)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	TmSTAT	–*	–*	–*	–*	–*	–*	TmRUN
Access	–	–	–	–	–	–	–	–	R	–	–	–	–	–	–	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

TMmCON1 is a special function register (SFR) used to control the Timer m.

Description of Bits

- **TmRUN** (bit 0)
  The TmRUN bit is used for controlling stop/start of timer m.
  In the 16-bit timer mode, be sure to set this bit to "0".

TmRUN	Description
0	Stops counting
1	Starts counting

- **TmSTAT** (bit 7)
  The TmSTAT bit is used for indicating "counting stopped"/"counting in progress" of timer m.
  In the 16-bit timer mode, this bit will read "0".

TmSTAT	Description
0	Counting stopped
1	Counting in progress

8.3 Description of Operation

8.3.1 Timer Mode Operation

The timer counters (TMnC) are set to an operating state (TnSTAT are set to "1") on the first falling edge of the timer clocks (TnCK) that are selected by TnCS bit of the Timer 0 to 7 control register 0 (TMnCON0) when the TnRUN bits of timer 0 to 7 control register 1 (TMnCON1) are set to "1" and increment the count value on the 2nd falling.

When the TMnC count value coincides the timer 0 to 7 data register (TMnD) value, a timer 0 to 7 interrupt (TMnINT) is generated on the next timer clock falling edge, TMnC is reset to "0x00", and the incremental counting continues.

When the TnRUN bit is set to "0", TMnC stops incremental counting after counting once the falling of the timer clock (TnCK). Confirm that TMnC has been stopped by checking that the TnSTAT bit of the Timer 0–7 control register 1 (TMnCON1) is "0".
When the TnRUN bits are set to "1" again, TMnC restart incremental counting from the previous values.
To initialize TMnC to "0x00", perform write operation in TMnC.

The timer interrupt period (T_{TMI}) is expressed by the following equation.

$$T_{TMI} = \frac{TMnD + 1}{TnCK\ (Hz)} \quad (n=0\ to\ 7)$$

TMnD: Timer 0 to 7 data register (TMnD) setting value (0x0000_0001 to 0x0000_00FF)
TnCK: Clock frequency selected by the Timer 0 to 7 control register 0 (TMnCON0)

After the TnRUN bit is set to "1", counting starts in synchronization with the timer clock. So, there may be an error of up to two clocks till the first timer interrupt. Subsequent timer interrupt periods are constant.
Figure 8-2 shows the operation timing diagram of Timer 0 to 7.

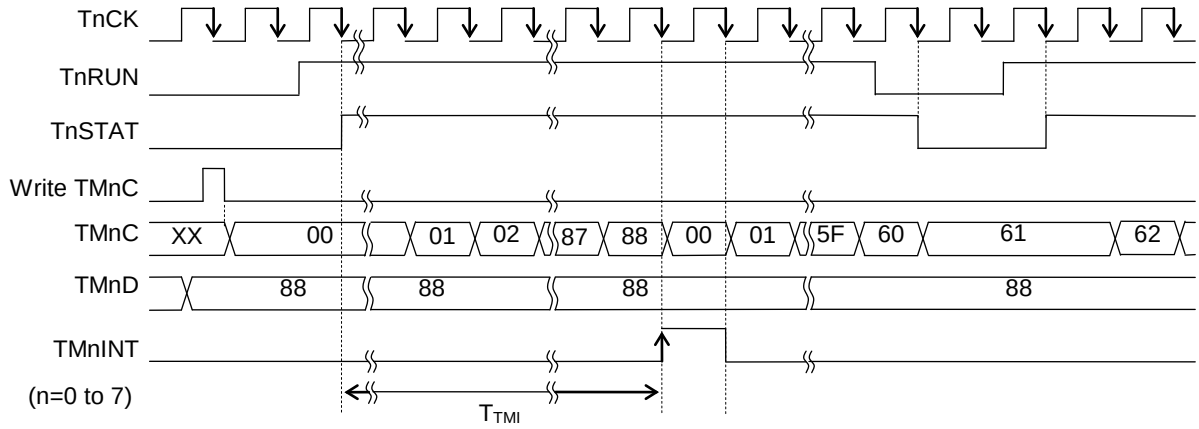


Figure 8-2 Operation Timing Diagram of Timer 0 to 7

[Note]
Even if "0" is written to the TnRUN bits, counting operation continues up to the falling edge (the timer 0 to 7 status flag (TnSTAT) is in a "1" state) of the next timer clock pulse. Therefore, the timer 0 to 7 interrupt (TMnINT) may occur.

8.4 Specifying Port Registers

To use the timer in the external clock mode, the applicable bit of each related port register needs to be set. See Chapter 15, "GPIO" for details about the port registers.

8.4.1 Operating Timer with External Clock (PWM)

Set PA0 (bit 1 and 0) of the PAMOD register to "10" to select the tertiary function of PA0.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**1**	**0**

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Set the bit 0 of the PIODIR register to "1" to set the input/output mode of PA0 to input.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR[6:0]						
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**1**

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the timer function
Input the operation clock for the timer from the PA0 pin.

[Note]
The PIOCON register does not need to be set. When the tertiary function is selected, the setting is automatically changed to the high-impedance input, but the value of PIOCON register does not change.

Chapter 9 PWM

9. PWM

9.1 Overview

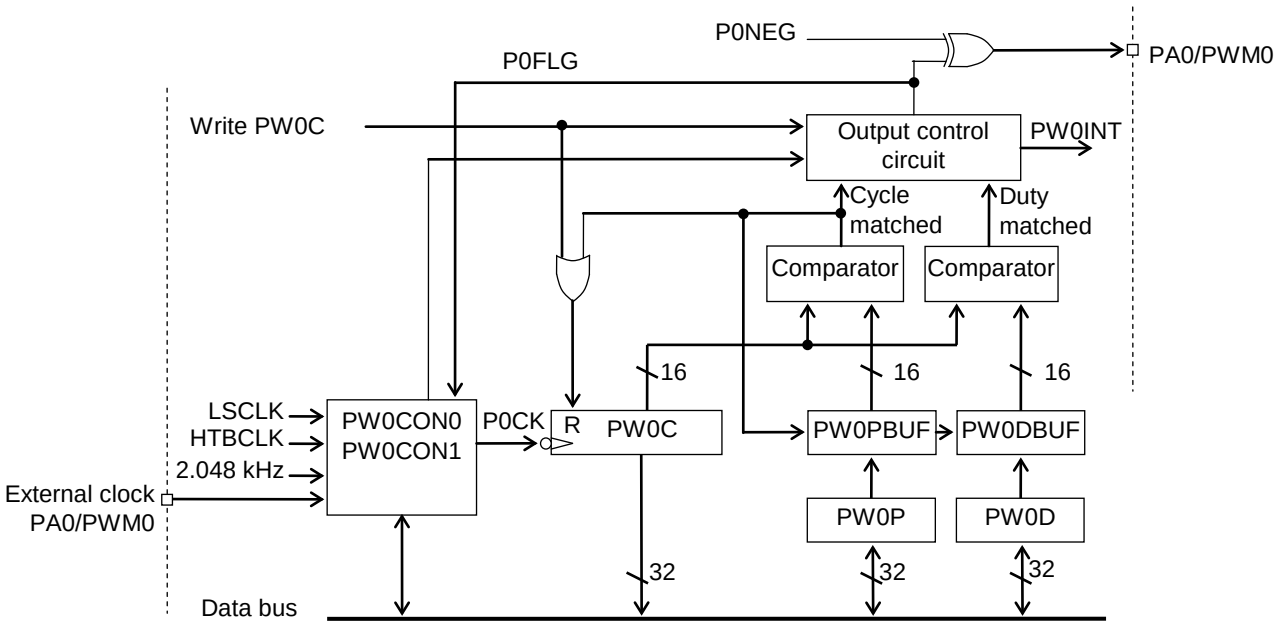
This LSI includes one channel of 16-bit PWM (Pulse Width Modulation).
The PWM output (PWM0) and PWM input (PWM0) are assigned to the tertiary function of PA0 (Port A). For functions of Port A, see Chapter 15, "GPIO".

9.1.1 Features

- A PWM signal using a clock of 250 ns (@HTBCLK = 4 MHz) to about 0.5 s (2.048 kHz) can be generated and output to the external.
- The output logic of the PWM signal can be switched to the positive or negative logic.
- At the coincidence of PWM signal period, duties, and period & duty, a PWM interrupt (PW0INT) occurs.
- For the PWM clock, a low-speed clock (LSCLK, 2.048 kHz), a high-speed time base clock (HTBCLK, 4 MHz to 250 kHz), and an external clock are available.

9.1.2 Configuration

Figure 9-1 shows the configuration of the PWM circuit.



PW0P : PWM0 cycle register
PW0PBUF : PWM0 period buffer
PW0D : PWM0 duty register
PW0DBUF : PWM0 duty buffer
PW0C : PWM0 counter register
PW0CON0 : PWM0 control register 0
PW0CON1 : PWM0 control register 1

Figure 9-1 Configuration of PWM Circuit

9.1.3 List of Pins

Pin name	I/O	Description
PA0/PWM0	I/O	Input setting: PWM0 external clock input pin Output setting: PWM0 output pin Used for the tertiary function of the PA0 pin.

9.2 Description of Registers

9.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4001_4400	PWM0 cycle register	PW0P	R/W	32	0x0000_FFFF
0x4001_4404	PWM0 duty register	PW0D	R/W	32	0x0000_0000
0x4001_4408	PWM0 counter register	PW0C	R/W	32	0x0000_0000
0x4001_440C	PWM0 control register 0	PW0CON0	R/W	32	0x0000_0000
0x4001_4410	PWM0 control register 1	PW0CON1	R/W	32	0x0000_0040

9.2.2 PWM0 Cycle Register (PW0P)

Address: 0x4001_4400
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	P0P[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
PW0P is a special function register (SFR) to set the PWM0 period.

[Note]
When PW0P is set to "0x0000", the PWM0 period buffer (PW0PBUF) is set to "0x0001".

9.2.3 PWM0 Duty Register (PW0D)

Address: 0x4001_4404
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	P0D[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
PW0D is a special function register (SFR) used to set the duty of PWM0.

[Note]
The PW0D data should be smaller than PW0P.

9.2.4 PWM0 Counter Register (PW0C)

Address: 0x4001_4408
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	P0C[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
PW0C is a special function register (SFR) that functions as 16-bit binary counter.
When a write operation is performed to PW0C, it is set to "0x0000". The data that is written is meaningless.

The content of PW0C during PWM operation cannot be read depending on the combination of the PWM clock and system clock.
Table 9-1 shows PW0C read enable/disable for each combination of the PWM clock and system clock.

Table 9-1 PW0C Read Enable/Disable during PWM Operation

PWM clock P0CK	System clock SYSCLK	PW0C read enable/disable
LSCLK, 2.048kHz	LSCLK	Read enabled.
LSCLK 2.048kHz	HSCLK	Read enabled. However, to prevent the reading of undefined data during counting up, read consecutively PW0C twice until the last data coincides the previous data.
HTBCLK	LSCLK	Read disabled.
HTBCLK	HSCLK	Read enabled.
External clock	LSCLK	Read disabled.
	HSCLK	

9.2.5 PWM0 Control Register 0 (PW0CON0)

Address: 0x4001_440C
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	P0NEG	P0IS[1:0]		P0CS[1:0]	
Access	-	-	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
PW0CON0 is a special function register (SFR) to control PWM.

[Description of Bits]

- **P0CS[1:0]** (bit 1, 0)
  The P0CS[1:0] is used to select the PWM0 operation clocks. LSCLK, HTBCLK, 2.048 kHz, or the external clock (PA0/PWM0) can be selected by these bits.

P0CS[1:0]		Description
0	0	LSCLK (initial value)
0	1	HTBCLK
1	0	2.048kHz
1	1	External clock (PA0/PWM0)

- **P0IS[1:0]** (bit 3, 2)
  The P0IS[1:0] is used to select the point at which the PWM0 interrupt occurs. "When the periods coincide", "when the duties coincide", or "when the periods and duties coincide" can be selected.

P0IS[1:0]		Description
0	0	When the periods coincide. (Initial value)
0	1	When the duties coincide.
1	*	When the periods and duties coincide.

- **P0NEG** (bit 4)
  The P0NEG bit is used to select the output logic of PWM0. The initial value of PWM0 output is "1" for the positive logic and "0" for the negative logic.

P0NEG	Description
0	Positive logic (initial value)
1	Negative logic

9.2.6 PWM0 Control Register 1 (PW0CON1)

Address: 0x4001_4410
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0040

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	P0STAT	P0FLG	-*	-*	-*	-*	-*	P0RUN
Access	-	-	-	-	-	-	-	-	R	R/W	-	-	-	-	-	R/W
Initial value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
PW0CON1 is a special function register (SFR) to control PWM0.

[Description of Bits]

- **P0RUN** (bit 0)
  The P0RUN bit is used to control count stop/start of PWM0.

P0RUN	Description
0	Stops counting. (Initial value)
1	Starts counting

- **P0FLG** (bit 6)
  The P0FLG bit is used to read the output flag of PWM0.
  When a write operation is performed to PW0C, it is set to "1".

P0FLG	Description
0	PWM0 output flag = "0"
1	PWM0 output flag = "1" (initial value)

- **P0STAT** (bit 7)
  The P0STAT bit indicates "counting stopped" or "counting in progress" of PWM0.

P0STAT	Description
0	Counting stopped. (Initial value)
1	Counting in progress.

9.3 Description of Operation

The PWM0 counter registers (PW0C) are set to an operating state (P0STAT is set to "1") on the first rising edge of the PWM clock (P0CK) that are selected by the PWM0 control register 0 (PW0CON0) when the P0RUN bit of PWM0 control register 1 (PW0CON1) is set to "1" and increment the count value on the 2nd rising edge.

When the count value of the PW0C coincides the value of the PWM0 duty buffer (PW0DBUF), the PWM flag (P0FLG) is set to "0" on the next rising edge of P0CK.
When the count value of the PW0C and the value of the PWM0 period buffer (PW0PBUF) coincide, the P0FLG becomes "1" at the next P0CK rising edge, and PW0C is reset to "0x0000" to continue incremental counting. At the same time, the value of the PWM0 duty register (PW0D) is transferred to the PWM0 duty buffer (PW0DBUF) and the value of PWM0 cycle register (PW0P) to the PWM0 period buffer (PW0PBUF).

When the P0RUN bit is set to "0", PW0C stops incremental counting after counting once the rising of the PWM clock (P0CK). Confirm that PW0C is stopped by checking that the P0STAT bit of the PWM0 control register 1 (PW0CON1) is "0". When the P0RUN bit is set to "1" again, the PW0C counter register restarts incremental counting from the previous value on the next rising edge of P0CK.

To initialize PW0C to "0x0000", perform write operation in PW0C. At that time, P0FLG is also set to "1".
During count stop (P0RUN is "0"), data written in the PWM0 duty register (PW0D) is transferred to the PWM0 duty buffer (PW0DBUF), and data written in the PWM0 cycle register (PW0P) is transferred to the PWM0 period buffer (PW0PBUF).

The PWM clock, the point at which an interrupt of PWM0 occurs, and the logic of the PWM output are selected by PWM0 control register 0 (PW0CON0).

The period of the PWM0 signal (T_{PWP}) and the first half duration (T_{PWD}) of the duty are expressed by the following equations.

$$T_{PWP} = \frac{PW0P + 1}{P0CK\ (Hz)}$$

$$T_{PWD} = \frac{PW0D + 1}{P0CK\ (Hz)}$$

PW0P: PWM0 cycle registers (PW0P) setting value (0x0001 to 0xFFFF)
PW0D: PWM0 duty registers (PW0D) setting value (0x0000 to 0xFFFE)
P0CK: Clock frequency selected by the PWM0 control register 0 (PW0CON0)

After the P0RUN bit is set to "1", counting starts in synchronization with the PWM clock. This causes an error of up to 1 clock pulse to the time the first PWM interrupt is issued. The PWM interrupt period from the second time is fixed.
Figure 9-2 shows the operation timing of PWM0.

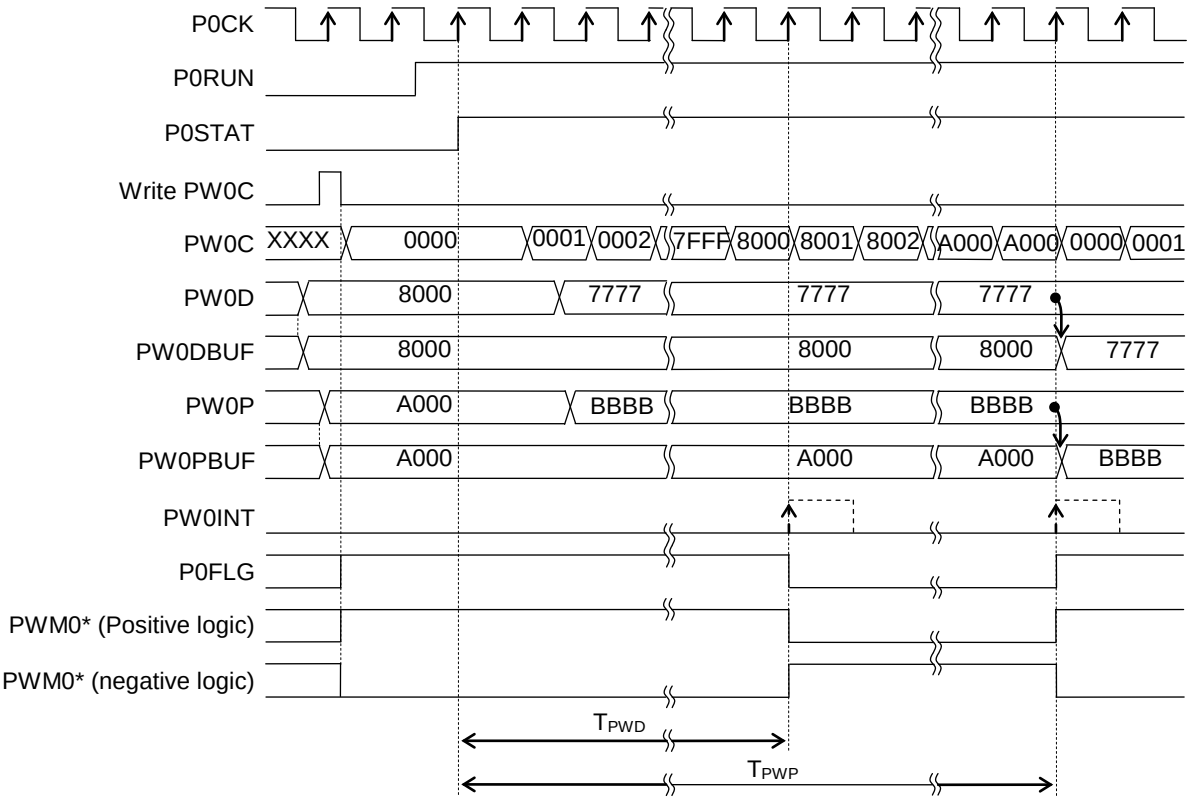


Figure 9-2 (1/2) Operation Timing Diagram of PWM0

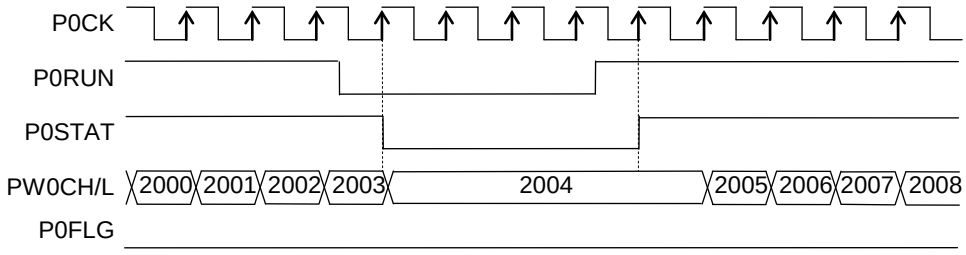


Figure 9-2 (2/2) Operation Timing Diagram of PWM0

[Note]
Even if "0" is written to the P0RUN bit, counting operation continues up to the rising edge (the PWM0 status flag (P0STAT) is in a "1" state) of the next PWM clock pulse. Therefore, the PWM0 interrupt (PW0INT) may occur.

9.4 Port Register Settings

To output the PWM waveform, the applicable bit of each related port register needs to be set. See Chapter 15, "GPIO" for details about the port registers.

9.4.1 Functioning PA0 Pin (PWM0) as PWM Output

Set PA0 (bit 1 and 0) of the PAMOD register to "10" to select the tertiary function of PA0.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**1**	**0**

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Set the bit 0 of the PIODIR register to "0" to set the input/output mode of PA0 to output.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR[6:0]						
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**0**

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the PWM function

[Note]
The PIOCON register does not need to be set. When the tertiary function is selected, the setting is automatically changed to the high-impedance output, but the value of PIOCON register does not change.

9.4.2 Operating PWM0 with External Clock (PWM)

Set PA0 (bit 1 and 0) of the PAMOD register to "10" to select the tertiary function of PA0.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**1**	**0**

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Set the bit 0 of the PIODIR register to "1" to set the input/output mode of PA0 to input.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR[6:0]						
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**1**

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the PWM function
Input the operation clock for the PWM0 from the PA0 pin.

[Note]
The PIOCON register does not need to be set. When the tertiary function is selected, the setting is automatically changed to the high-impedance input, but the value of PIOCON register does not change.

Chapter 10 Watchdog Timer

10. Watchdog Timer

10.1 Overview

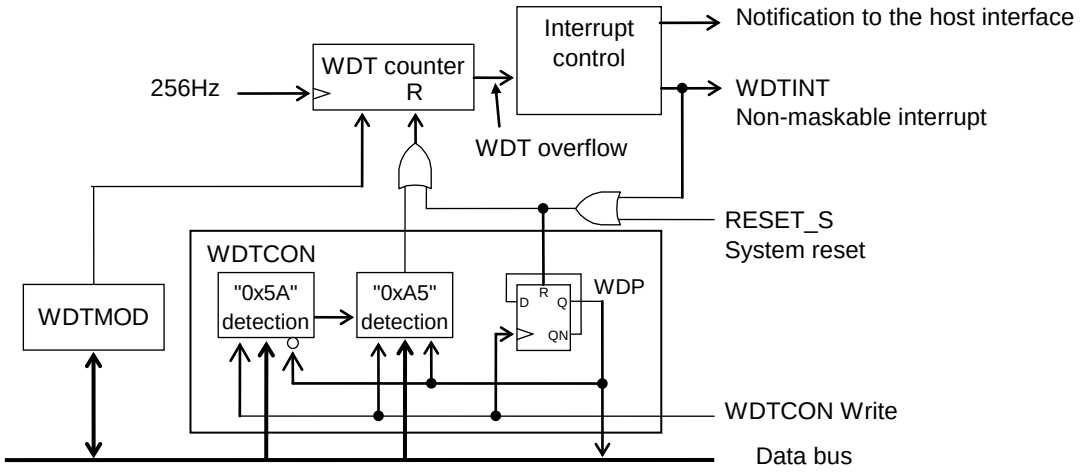
This LSI includes a watchdog timer (WDT) which operates in system reset mode unconditionally (free-run operation) in order to detect an undefined state of the MCU.
If the WDT counter overflows due to the failure of clearing of the WDT counter within the WDT overflow period, the watchdog timer requests a WDT interrupt (non-maskable interrupt). If the second overflow occurs, it is determined as an abnormal state. If an abnormal state is detected, the host interface is notified of it. The host CPU should reset this LSI when it receives the abnormal state.
See Chapter 7, "Interrupt" for interrupts and Chapter 11, "Host Interface" for abnormal state notification.

10.1.1 Features

- Free running (cannot be stopped)
- One of four types of overflow periods (125ms, 500ms, 2s, and 8s) selectable by software
- Non-maskable interrupt requested by overflow
- Operation at the second overflow
  -Notification of abnormal state to the host interface

10.1.2 Configuration

Figure 10-1 shows the configuration of the watchdog timer.



WDTCON : Watchdog timer control register
WDTMOD : Watchdog timer mode register

Figure 10-1 Configuration of Watchdog Timer

10.2 Description of Registers

10.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4001_0400	Watchdog timer control register	WDTCON	R/W	32	0x0000_0000
0x4001_0404	Watchdog timer mode register	WDTMOD	R/W	32	0x0000_0002

10.2.2 Watchdog Timer Control Register (WDTCON)

Address: 0x4001_0400
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	d7	d6	d5	d4	d3	d2	d1	WDP/ d0
Access	-	-	-	-	-	-	-	-	W	W	W	W	W	W	W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
WDTCON is a special function register (SFR) to clear the WDT counter.
When WDTCON is read, the value of the internal pointer (WDP) is read from bit 0.

[Description of Bits]

- **WDP/d0** (bit 0)
  The value of the internal pointer (WDP) is read from this bit. The WDP is reset to "0" at the system reset or WDT counter overflow and is inverted every writing to WDTCON.

- **d7-d0** (bit 7-0)
  This bit is used to write data to clear the WDT counter. The WDT counter can be cleared by writing "0x5A" with the internal pointer (WDP) is "0", then writing "0xA5" with the WDP "1".

[Note]
When the WDT interrupt (WDTINT) occurs by the first overflow of WDT counter, the WDT counter and internal pointer (WDP) are initialized for the half of low-speed clock (about 15 µs). Then, writing to WDTCON during this period is invalid and WDP does not invert. Therefore, in the case of that you have program codes handle to clear the WDT when the first overflow WDT interrupt occurs and also the codes run at high-speed system clock, please check the WDP gets reversed after writing to WDTCON to see if the writing was surely successful. For example of the program code, see Section "10.4 Example of Processing When Not Using Watchdog Timer".

10.2.3 Watchdog Timer Mode Register (WDTMOD)

Address: 0x4001_0404
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	WDT[1:0]	
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
WDTMOD is a special function register (SFR) to set the overflow period of the WDT counter.

[Description of Bits]

- **WDT[1:0]** (bit 1-0)
  These bits are used to select an overflow period of the watchdog timer.
  The WDT[1:0] is set an overflow period (T_{WOV}) of the WDT counter. One of 125ms, 500ms, 2s, and 8s can be selected.

WDT[1:0]		Description
0	0	125ms
0	1	500ms
1	0	2s (initial value)
1	1	8s

10.3 Description of Operation

The WDT counter starts incremental counting after the system reset has been released and the low-speed clock (LSCLK) oscillation start.
The WDT counter can be cleared by writing "0x5A" with the internal pointer (WDP) is "0", then writing "0xA5" with the WDP "1".
The WDP is reset to "0" at the system reset or WDT counter overflow and is inverted every writing to WDTCON.
When the WDT counter cannot be cleared within the WDT counter overflow period (T_{WOV}), a watchdog timer interrupt (WDTINT) occurs.
For the overflow period (T_{WOV}) of the WDT counter, one of 125ms, 500ms, 2s, and 8s can be selected by the watchdog timer mode register (WDTMOD).
Clear the WDT counter within the clear period of the WDT counter (T_{WCL}) shown in Table 10-1.

Table 10-1 Clear Period of WDT Counter

WDT[1:0]		T_{WOV}	T_{WCL}
0	0	125ms	Approximately 121ms
0	1	500ms	Approx. 496 ms
1	0	2000ms	Approx. 1996 ms
1	1	8000ms	Approx. 7996 ms

Figure 10-2 shows an example of watchdog timer operation.

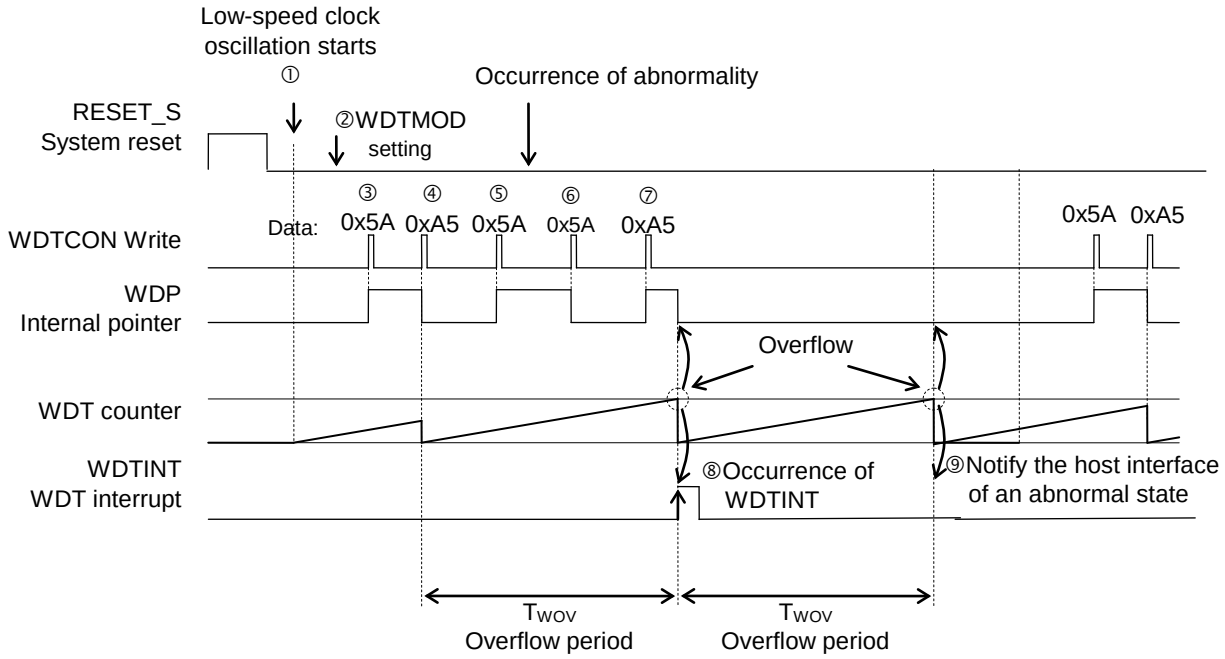


Figure 10-2 Example of Watchdog Timer Operation

① When the system reset is released and the low-speed clock (LSCLK) is output, the WDT counter starts incremental counting.
② The overflow period of the WDT counter (T_{WOV}) is set to WDTMOD.
③ "0x5A" is written to WDTCON. (Internal pointer 0 to 1)
④ "0xA5" is written to WDTCON and the WDT counter is cleared. (Internal pointer 1 to 0)
⑤ "0x5A" is written to WDTCON. (Internal pointer 0 to 1)
⑥ When "0x5A" is written to WDTCON after the occurrence of abnormality, it cannot be accepted as the internal pointer is set to "1". (Internal pointer 1 to 0)
⑦ Although "0xA5" is written to WDTCON, the WDT counter is not cleared since the internal pointer is "0" and the writing of "0x5A" is not accepted in ⑥. (Internal pointer 0 to 1)
⑧ The WDT counter overflows and a watchdog timer interrupt request (WDTINT) is generated. At this point, the WDT counter and internal pointer are initialized for the half of low-speed clock (about 15.26 µs). (Internal pointer 1 to 0)
⑨ When the WDT counter overflows again after the WDTINT, the host interface is notified of the abnormal state.

[Note]
- In the sleep mode, the watchdog timer operation does not stop. If WDT interrupt occurs during the sleep mode, the sleep mode is released.
- The watchdog timer cannot detect all the abnormal operations. Even if the CPU loses control, the watchdog timer cannot detect the abnormality in the operation state in which the WDT counter is cleared.

10.4 Example of Processing When Not Using Watchdog Timer

The WDT counter is a free running counter that starts incremental counting unconditionally when the system reset is released and the low-speed clock (LSCLK) is output. Since the WDT counter overflow causes a non-maskable interrupt, the clearance process of WDT counter needs to be performed even when the WDT function is not used as fail safe.
Figure 10-3 shows an example of program that clears the WDT counter by the WDT interrupt routine.

```
__disable_irq();
do
{
    WDT->WDTCON = 0x5a;
} while(WDT->WDTCON&0x1 != 1);
WDT->WDTCON = 0xa5;
__enable_irq();
```

Figure 10-3 Example of Program Description

Chapter 11 Host Interface

11. Host Interface

11.1 Overview

To communicate with the host processor (such as application processors), this LSI includes a host interface that has a SPI or I2C interface and interrupt and register functions.

For interrupts to the CPU, see Chapter 7, "Interrupt."

11.1.1 Features

- SPI or I2C can be selected as an interface with the host processor.
  For SPI, four-wire or three-wire can be selected. And SCS polarity can be selected.
  For four-wire, HiZ or Low output can be selected as a data output signal in the no output mode.
- Controls the interrupt signal to the host processor. Two interrupt signals can be controlled individually as usage.
- WDT overflow causes a notification of abnormal state to the host.
- When a command is input by the host processor, a CPU interrupt occurs allowing the CPU to receive the command.

[Note]
As for SPI, if Host keeps SCLK high while communication is standby, there are cases when this LSI cannot enter SleepDeep mode until SCS_S becomes non-active.

11.1.2 Configuration

Figure 11-1 shows the configuration of the host interface.

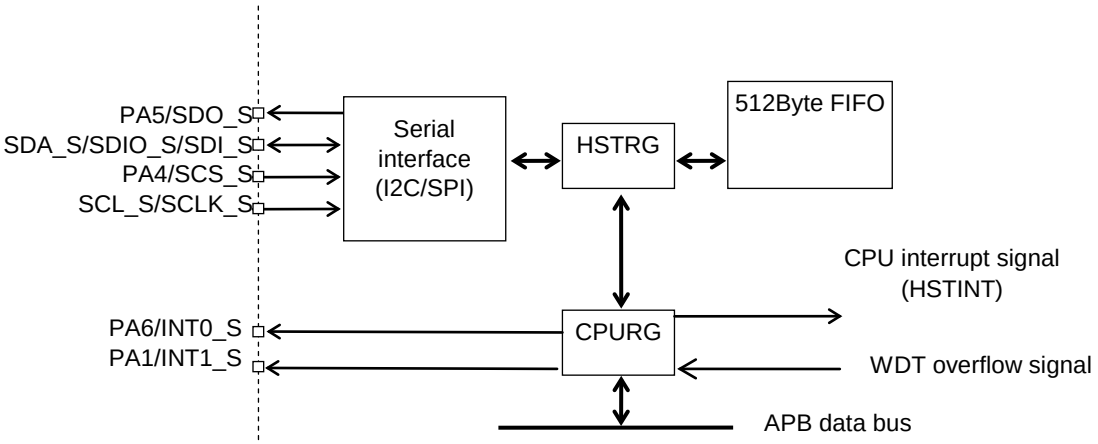


Figure 11-1 Configuration of Host Interface

11.2 Description of Registers

11.2.1 Register List (CPURG: for CPU Access)

Base address: 0x4005_0000

Address	Name	Symbol (Word)	Symbol (Half-Word)	Symbol (Byte)	R/W	Size [bits]	Initial value
0x00	Configuration register	HIFCFG	-	-	R	32	0xFFFF_2F00
0x04	Reserved	-	-	-	-	-	-
0x08	Operation status register	HIFST	-	-	R/W	32	0x0000_FE00
0x0C	Interrupt request register	HIFRQ	-	-	R/W	32	0x0000_0000
0x10	FIFO register	HIFFIFOW	-	HIFFIFO	R/W	32/8	0x0000_00XX
0x14	FIFO switch register	HIFFSEL	-	-	R/W	32	0x0000_0000
0x18	FIFO write pointer register	HIFWP	-	-	R/W	32	0x0000_0000
0x1C	FIFO read pointer register	HIFRP	-	-	R/W	32	0x0000_0000
0x20	Parameter register F	HIFPRMF	-	-	R	32	0x0000_0000
0x24	Parameter register B	HIFPRMB	-	-	R	32	0x0000_0000
0x28	Parameter register 7	HIFPRM7	-	-	R	32	0x0000_0000
0x2C	Parameter register 3	HIFPRM3	-	-	R	32	0x0000_0000
0x30	Command register	HIFCMD	-	-	R/W	32	0x0000_0000
0x34 – 0x3C	Reserved	-	-	-	-	-	-
0x40	Result register 00	HIFRLT00W	HIFRLT00H	HIFRLT00	R/W	32/16/8	0x00
0x41	Result register 01	-	-	HIFRLT01	R/W	8	0x00
0x42	Result register 02	-	HIFRLT02H	HIFRLT02	R/W	16/8	0x00
0x43	Result register 03	-	-	HIFRLT03	R/W	8	0x00
0x44	Result register 04	HIFRLT04W	HIFRLT04H	HIFRLT04	R/W	32/16/8	0x00
0x45	Result register 05	-	-	HIFRLT05	R/W	8	0x00
0x46	Result register 06	-	HIFRLT06H	HIFRLT06	R/W	16/8	0x00
0x47	Result register 07	-	-	HIFRLT07	R/W	8	0x00
0x48	Result register 08	HIFRLT08W	HIFRLT08H	HIFRLT08	R/W	32/16/8	0x00
0x49	Result register 09	-	-	HIFRLT09	R/W	8	0x00
0x4A	Result register 0A	-	HIFRLT0AH	HIFRLT0A	R/W	16/8	0x00
0x4B	Result register 0B	-	-	HIFRLT0B	R/W	8	0x00
0x4C	Result register 0C	HIFRLT0CW	HIFRLT0CH	HIFRLT0C	R/W	32/16/8	0x00
0x4D	Result register 0D	-	-	HIFRLT0D	R/W	8	0x00
0x4E	Result register 0E	-	HIFRLT0EH	HIFRLT0E	R/W	16/8	0x00
0x4F	Result register 0F	-	-	HIFRLT0F	R/W	8	0x00
0x50	Result register 10	HIFRLT10W	HIFRLT10H	HIFRLT10	R/W	32/16/8	0x00
0x51	Result register 11	-	-	HIFRLT11	R/W	8	0x00
0x52	Result register 12	-	HIFRLT12H	HIFRLT12	R/W	16/8	0x00
0x53	Result register 13	-	-	HIFRLT13	R/W	8	0x00
0x54	Result register 14	HIFRLT14W	HIFRLT14H	HIFRLT14	R/W	32/16/8	0x00
0x55	Result register 15	-	-	HIFRLT15	R/W	8	0x00
0x56	Result register 16	-	HIFRLT16H	HIFRLT16	R/W	16/8	0x00
0x57	Result register 17	-	-	HIFRLT17	R/W	8	0x00
0x58	Result register 18	HIFRLT18W	HIFRLT18H	HIFRLT18	R/W	32/16/8	0x00
0x59	Result register 19	-	-	HIFRLT19	R/W	8	0x00
0x5A	Result register 1A	-	HIFRLT1AH	HIFRLT1A	R/W	16/8	0x00
0x5B	Result register 1B	-	-	HIFRLT1B	R/W	8	0x00
0x5C	Result register 1C	HIFRLT1CW	HIFRLT1CH	HIFRLT1C	R/W	32/16/8	0x00
0x5D	Result register 1D	-	-	HIFRLT1D	R/W	8	0x00
0x5E	Result register 1E	-	HIFRLT1EH	HIFRLT1E	R/W	16/8	0x00
0x5F	Result register 1F	-	-	HIFRLT1F	R/W	8	0x00
0x60	Result register 20	HIFRLT20W	HIFRLT20H	HIFRLT20	R/W	32/16/8	0x00
0x61	Result register 21	-	-	HIFRLT21	R/W	8	0x00

Address	Name	Symbol (Word)	Symbol (Half-Word)	Symbol (Byte)	R/W	Size [bits]	Initial value
0x62	Result register 22	-	HIFRLT22H	HIFRLT22	R/W	16/8	0x00
0x63	Result register 23	-	-	HIFRLT23	R/W	8	0x00
0x64	Result register 24	HIFRLT24W	HIFRLT24H	HIFRLT24	R/W	32/16/8	0x00
0x65	Result register 25	-	-	HIFRLT25	R/W	8	0x00
0x66	Result register 26	-	HIFRLT26H	HIFRLT26	R/W	16/8	0x00
0x67	Result register 27	-	-	HIFRLT27	R/W	8	0x00
0x68	Result register 28	HIFRLT28W	HIFRLT28H	HIFRLT28	R/W	32/16/8	0x00
0x69	Result register 29	-	-	HIFRLT29	R/W	8	0x00
0x6A	Result register 2A	-	HIFRLT2AH	HIFRLT2A	R/W	16/8	0x00
0x6B	Result register 2B	-	-	HIFRLT2B	R/W	8	0x00
0x6C	Result register 2C	HIFRLT2CW	HIFRLT2CH	HIFRLT2C	R/W	32/16/8	0x00
0x6D	Result register 2D	-	-	HIFRLT2D	R/W	8	0x00
0x6E	Result register 2E	-	HIFRLT2EH	HIFRLT2E	R/W	16/8	0x00
0x6F	Result register 2F	-	-	HIFRLT2F	R/W	8	0x00
0x70	Result register 30	HIFRLT30W	HIFRLT30H	HIFRLT30	R/W	32/16/8	0x00
0x71	Result register 31	-	-	HIFRLT31	R/W	8	0x00
0x72	Result register 32	-	HIFRLT32H	HIFRLT32	R/W	16/8	0x00
0x73	Result register 33	-	-	HIFRLT33	R/W	8	0x00
0x74	Result register 34	HIFRLT34W	HIFRLT34H	HIFRLT34	R/W	32/16/8	0x00
0x75	Result register 35	-	-	HIFRLT35	R/W	8	0x00
0x76	Result register 36	-	HIFRLT36H	HIFRLT36	R/W	16/8	0x00
0x77	Result register 37	-	-	HIFRLT37	R/W	8	0x00
0x78	Result register 38	HIFRLT38W	HIFRLT38H	HIFRLT38	R/W	32/16/8	0x00
0x79	Result register 39	-	-	HIFRLT39	R/W	8	0x00
0x7A	Result register 3A	-	HIFRLT3AH	HIFRLT3A	R/W	16/8	0x00
0x7B	Result register 3B	-	-	HIFRLT3B	R/W	8	0x00
0x7C	Result register 3C	HIFRLT3CW	HIFRLT3CH	HIFRLT3C	R/W	32/16/8	0x00
0x7D	Result register 3D	-	-	HIFRLT3D	R/W	8	0x00
0x7E	Result register 3E	-	HIFRLT3EH	HIFRLT3E	R/W	16/8	0x00
0x7F	Result register 3F	-	-	HIFRLT3F	R/W	8	0x00

Accessing any of reserved areas is prohibited. Operation cannot be guaranteed if you access a reserved area.

11.2.2 Register List (HSTRG: for Host Access)

Address		Name	Symbol	R/W	Size [bits]	Initial value
Write	Read					
0x00	0x80	Configuration register	CFG	R/W	8	0x00
0x01	0x81	Reserved	-	-	-	-
0x02	0x82	Interrupt mask register 0	INTMSK0	R/W	8	0xFF
0x03	0x83	Interrupt mask register 1	INTMSK1	R/W	8	0xFF
0x04~ 0x08	0x84~ 0x88	Reserved	-	-	-	-
-	0x89	Operation status register	STATUS	R	8	0xFE
-	0x8A	Error code register 0	ERROR0	R	8	0x00
-	0x8B	Error code register 1	ERROR1	R	8	0x00
-	0x8C	Interrupt request register 0	INTREQ0	R	8	0x00
-	0x8D	Interrupt request register 1	INTREQ1	R	8	0x00
0x0E~ 0x0F	0x8E~ 0x8F	Reserved	-	-	-	-
0x10	0x90	FIFO register	FIFO	R/W	8	0xXX
0x11~ 0x1F	0x91~ 0x9F	Reserved	-	-	-	-
0x20	0xA0	Parameter register 0F	PRMF	R/W	8	0x00
0x21	0xA1	Parameter register 0E	PRME	R/W	8	0x00
0x22	0xA2	Parameter register 0D	PRMD	R/W	8	0x00
0x23	0xA3	Parameter register 0C	PRMC	R/W	8	0x00
0x24	0xA4	Parameter register 0B	PRMB	R/W	8	0x00
0x25	0xA5	Parameter register 0A	PRMA	R/W	8	0x00
0x26	0xA6	Parameter register 09	PRM9	R/W	8	0x00
0x27	0xA7	Parameter register 08	PRM8	R/W	8	0x00
0x28	0xA8	Parameter register 07	PRM7	R/W	8	0x00
0x29	0xA9	Parameter register 06	PRM6	R/W	8	0x00
0x2A	0xAA	Parameter register 05	PRM5	R/W	8	0x00
0x2B	0xAB	Parameter register 04	PRM4	R/W	8	0x00
0x2C	0xAC	Parameter register 03	PRM3	R/W	8	0x00
0x2D	0xAD	Parameter register 02	PRM2	R/W	8	0x00
0x2E	0xAE	Parameter register 01	PRM1	R/W	8	0x00
0x2F	0xAF	Parameter register 00	PRM0	R/W	8	0x00
0x30	0xB0	Command register 0	CMD0	R/W	8	0x00
0x31	0xB1	Command register 1	CMD1	R/W	8	0x00
0x32	0xB2	Command entry register	ENT	R/W	8	0x00
0x33~ 0x3F	0xB3~ 0xBF	Reserved	-	-	-	-
-	0xC0	Result register 00	RSLT00	R	8	0x00
-	0xC1	Result register 01	RSLT01	R	8	0x00
-	0xC2	Result register 02	RSLT02	R	8	0x00
-	0xC3	Result register 03	RSLT03	R	8	0x00
-	0xC4	Result register 04	RSLT04	R	8	0x00
-	0xC5	Result register 05	RSLT05	R	8	0x00
-	0xC6	Result register 06	RSLT06	R	8	0x00
-	0xC7	Result register 07	RSLT07	R	8	0x00
-	0xC8	Result register 08	RSLT08	R	8	0x00
-	0xC9	Result register 09	RSLT09	R	8	0x00
-	0xCA	Result register 0A	RSLT0A	R	8	0x00
-	0xCB	Result register 0B	RSLT0B	R	8	0x00
-	0xCC	Result register 0C	RSLT0C	R	8	0x00
-	0xCD	Result register 0D	RSLT0D	R	8	0x00
-	0xCE	Result register 0E	RSLT0E	R	8	0x00

Address		Name	Symbol	R/W	Size [bits]	Initial value
Write	Read					
-	0xCF	Result register 0F	RSLT0F	R	8	0x00
-	0xD0	Result register 10	RSLT10	R	8	0x00
-	0xD1	Result register 11	RSLT11	R	8	0x00
-	0xD2	Result register 12	RSLT12	R	8	0x00
-	0xD3	Result register 13	RSLT13	R	8	0x00
-	0xD4	Result register 14	RSLT14	R	8	0x00
-	0xD5	Result register 15	RSLT15	R	8	0x00
-	0xD6	Result register 16	RSLT16	R	8	0x00
-	0xD7	Result register 17	RSLT17	R	8	0x00
-	0xD8	Result register 18	RSLT18	R	8	0x00
-	0xD9	Result register 19	RSLT19	R	8	0x00
-	0xDA	Result register 1A	RSLT1A	R	8	0x00
-	0xDB	Result register 1B	RSLT1B	R	8	0x00
-	0xDC	Result register 1C	RSLT1C	R	8	0x00
-	0xDD	Result register 1D	RSLT1D	R	8	0x00
-	0xDE	Result register 1E	RSLT1E	R	8	0x00
-	0xDF	Result register 1F	RSLT1F	R	8	0x00
-	0xE0	Result register 20	RSLT20	R	8	0x00
-	0xE1	Result register 21	RSLT21	R	8	0x00
-	0xE2	Result register 22	RSLT22	R	8	0x00
-	0xE3	Result register 23	RSLT23	R	8	0x00
-	0xE4	Result register 24	RSLT24	R	8	0x00
-	0xE5	Result register 25	RSLT25	R	8	0x00
-	0xE6	Result register 26	RSLT26	R	8	0x00
-	0xE7	Result register 27	RSLT27	R	8	0x00
-	0xE8	Result register 28	RSLT28	R	8	0x00
-	0xE9	Result register 29	RSLT29	R	8	0x00
-	0xEA	Result register 2A	RSLT2A	R	8	0x00
-	0xEB	Result register 2B	RSLT2B	R	8	0x00
-	0xEC	Result register 2C	RSLT2C	R	8	0x00
-	0xED	Result register 2D	RSLT2D	R	8	0x00
-	0xEE	Result register 2E	RSLT2E	R	8	0x00
-	0xEF	Result register 2F	RSLT2F	R	8	0x00
-	0xF0	Result register 30	RSLT30	R	8	0x00
-	0xF1	Result register 31	RSLT31	R	8	0x00
-	0xF2	Result register 32	RSLT32	R	8	0x00
-	0xF3	Result register 33	RSLT33	R	8	0x00
-	0xF4	Result register 34	RSLT34	R	8	0x00
-	0xF5	Result register 35	RSLT35	R	8	0x00
-	0xF6	Result register 36	RSLT36	R	8	0x00
-	0xF7	Result register 37	RSLT37	R	8	0x00
-	0xF8	Result register 38	RSLT38	R	8	0x00
-	0xF9	Result register 39	RSLT39	R	8	0x00
-	0xFA	Result register 3A	RSLT3A	R	8	0x00
-	0xFB	Result register 3B	RSLT3B	R	8	0x00
-	0xFC	Result register 3C	RSLT3C	R	8	0x00
-	0xFD	Result register 3D	RSLT3D	R	8	0x00
-	0xFE	Result register 3E	RSLT3E	R	8	0x00
-	0xFF	Result register 3F	RSLT3F	R	8	0x00

Accessing any of reserved areas is prohibited. Operation cannot be guaranteed if you access a reserved area.

11.2.3 CPURG: Configuration Register (HIFCFG)

Address: 0x4005_0000
Access: R/W
Access size: 32 Bits
Initial value: 0xFFFF_2F00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	MSK1[7:0]								MSK0[7:0]							
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	IFCFG[7:1]							IF SEL	REG MD	–*	INTPW[1:0]		INT1 EN	INT LVL	–*	–*
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	–	R	R	R	R	–	–
Initial value	0	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
Configuration register (HIFCFG) is a special function register (SFR) used to indicate the configuration that was set to the register for host access (HSTRG:CFG) by the host processor and to set the serial interface.

[Description of Bits]

- **INTLVL** (bit 2)

This bit indicates whether the interrupt signal to the host processor is a level output or pulse output.

INTLVL	Description
0	Pulse output (initial value)
1	Level output

This bit is common to INT0_S and INT1_S.
For pulse output, the time set in INTPW[1:0] is used as the pulse width.

- **INT1EN** (bit 3)

This bit indicates whether the second interrupt signal to the host processor is allowed or not.

INT1EN	Description
0	INT1_S is merged with INT0_S to be output (initial value)
1	INT1_S is output independently

[Note] Separately set the port (PA1) to output INT1_S.

- **INTPW** (bit 5-4)

This bit indicates the pulse width setting when the interrupt signal is a pulse signal.

INTPW[1:0]	Description
00	250[ns] (4 MHz cycle) (initial value)
01	500[ns] (2 MHz cycle)
10	1000[ns] (1 MHz cycle)
11	2000[ns] (500 kHz cycle)

- **REGMD** (bit 7)
  This bit shows the register access mode of the serial interface (SPI/I2C). When set to "0", the internal address is incremented by 1 each time a 1-byte data is transmitted/received. When set to "1", the address is fixed to the same address.

REGMD	Description
0	Address increment enabled (initial value)
1	Address increment disabled

  FIFO register (FIFO) and result register 3F(RSLT3F) are excluded from the address increment. If FIFO register or result register 3F is accessed with this bit set to "0", the address increment is not executed.

- **IFSEL** (bit 8)
  This bit indicates whether the serial interface is I2C or SPI.

IFSEL	Description
0	SPI interface
1	I2C interface (initial value)

- **IFCFG** (bit 15-9)
  The function of this bit depends on the IFSEL setting. This bit should be set after setting IFSEL.
  IFSEL=1 (I2C selected): Indicates the I2C slave address. The initial value of the I2C slave address is 0x17. When using I2C, set the I2C slave address from the CPU.
  IFSEL=0 (SPI selected): The function of each bit is shown below.

IFCFG[7]	Description
0	Sets SPI slave to three-wire (initial value)
1	Sets SPI slave to four-wire

IFCFG[6]	Description
0	SPI output data is Hi-Z (initial value) in the no output mode
1	SPI output data is 0 in the no output mode

IFCFG[5]	Description
0	CS of SPI is "0" active
1	CS of SPI is "1" active (initial value)

- **MSK1,0** (bit 31-16)
  This bit masks interrupt sources due to the interrupt request register.

MSK0[n] MSK1[n]	Description
0	Does not mask the interrupt source due to the HSTRG:INTREQ0 REQ0[n] bit
1	Masks the interrupt source due to the HSTRG:INTREQ0 REQ0[n] bit (initial value)

  n=0 to 7

11.2.4 CPURG: Operation Status Register (HIFST)

Address: 0x4005_0008
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_FE00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	ER1[7:0]								ER0[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	ST[7:0]								–*	–*	–*	–*	–*	–*	RLTP	INTP
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	–	–	–	–	–	–	R	R
Initial value	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
Operation status register (HIFST) is a special function register (SFR) used to indicate the sensor measurement status and notify the host processor of the error code.

[Description of Bits]

- **ST** (bit 15-8)
  This bit indicates the status of the sensor measurement. When this bit is set, the set value is transferred to the register for host access (HSTRG:STATUS). For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification".

- **ER1, ER0** (bit 31-16)
  This bit sets the interrupt source of the error code to the host processor. When this bit is set, the set value is transferred to the register for host access (HSTRG:ERROR0,1). For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

- **RLTP** (bit 1)
  This bit indicates the write status to the result register.
  Confirm that this bit is set to "0" before power-down.
  In case of power-down with this bit set to "1", the written data is reflected to the result register after returning from power-down.

RLTP	Description
0	No writing to result register (initial value)
1	Writing to result register

- **INTP** (bit 0)
  This bit indicates the write status to the interrupt request register.
  Confirm that this bit is set to "0" before power-down after issuing an interrupt request to the host processor.
  In case of power-down with this bit set to "1", the written data is reflected to the interrupt request register after returning from power-down. Therefore, an interrupt signal to the host is not asserted before returning from power-down.

INTP	Description
0	No writing to interrupt request register (initial value)
1	Writing to interrupt request register

11.2.5 CPURG: Interrupt Request Register (HIFRQ)

Address: 0x4005_000C
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name				REQ1[7:0]								REQ0[7:0]				
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
Interrupt request register (HIFRQ) is a special function register (SFR) used to notify the host processor of the interrupt request.
When this register is set, the set value is transferred to the register for host access (HSTRG:INTREQ0,1) causing an interrupt for the host processor. The operation of interrupt differs depending on the INTLVL bit of configuration register (HSTRG:CFG).

When INTLVL=1 (level output), an interrupt signal is output to the host processor if the either bit is set to "1" and the sensor interrupt mask 0/1 (HSTRG: INTMSK0/1) of that bit allows it. Then, after the host processor reads and clears all the interrupt sources, the interrupt signal to the host processor will also be cleared.

When INTLVL=0 (pulse output), an interrupt of one cycle is output to the host processor at the next cycle if "1" is set to the "0"-cleared bit and the sensor interrupt mask 0/1 (HSTRG: INTMSK0/1) of that bit allows it. Setting "1" to a bit that has been set to "1" does not cause an interrupt.

However, if a WDT overflow signal is input (if the WDTP of WDT is not cleared and WDT overflow occurs), 0x0000_FFFF is read until the hardware reset is performed.

[Description of Bits]

- **REQ1,0** (bit 15-0)

  Sets the interrupt source to the host processor. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

  CPU can write only "1" to this register. "Writing 0" is ignored.
  Only read access from the host can clear this register (to "0").

  If CPU writes "1" to any bit with REQ1 or REQ0 bit set to "1", an interrupt pulse is output at the time of writing when INTLVL=0 (pulse output).
  The interrupt level is held when INTLVL=1 (level output).

  The timing of clearing REQ1,0 depends on the INT1EN bit setting of the configuration register (CPURG:HIFCFG). For details, see "11.4.5Timing of Clearing Interrupt Request Register".

11.2.6 CPURG: FIFO Register (HIFFIFO)

Address: 0x4005_0010
Access: R/W
Access size: 8/32 Bits
Initial value: 0x0000_00XX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	FIFO[7:0]							
Access	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
FIFO register (HIFFIFO) is a special function register (SFR) consisting of 512-byte FIFO.
CPU can access this register when FSEL=0. CPU cannot access it when FSEL=1.

[Description of Bits]
- **FIFO** (bit 7-0)
  This bit shows the FIFO data. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.7 CPURG: FIFO Switch Register (HIFFSEL)

Address: 0x4005_0014
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	FSEL
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
FIFO switch register (HIFFSEL) is a special function register (SFR) used to switch whether host processor or CPU can access 512-byte FIFO.

[Description of Bits]
- **FSEL** (bit 0)

FSEL	Description
0	Only CPU can access FIFO (initial value)
1	Only host processor can access FIFO

11.2.8 CPURG: FIFO Write Pointer Register (HIFWP)

Address: 0x4005_0018
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	FIFOWP[8:0]								
Access	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
FIFO write pointer register (HIFWP) is a special function register (SFR) that has the pointer to specify the address to write to the FIFO.
This register can be rewritten by the CPU only when FSEL=0 (CPU access enabled).
Rewriting by the CPU is disabled when FSEL=1.

[Description of Bits]
- **FIFOWP** (bit 8-0)
  The pointer to specify the address to write to the FIFO. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.9 CPURG: FIFO Read Pointer Register (HIFRP)

Address: 0x4005_001C
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	FIFORP[8:0]								
Access	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
FIFO read pointer register (HIFRP) is a special function register (SFR) that has the pointer to specify the address to read from the FIFO.
This register can be rewritten by the CPU only when FSEL=0 (CPU access enabled).
Rewriting by the CPU is disabled when FSEL=1.

[Description of Bits]
- **FIFORP** (bit 8-0)
  The pointer to specify the address to read from the FIFO. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.10 CPURG: Parameter Register (HIFPRMF, HIFPRMB, HIFPRM7, HIFPRM3)

Address: 0x4005_0020 to 0x4005_002C
Access: R
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	PRMC[7:0]								PRMD[7:0]							
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	PRME[7:0]								PRMF[7:0]							
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
Parameter register (IFPRMF, HIFPRMB, HIFPRM7, HIFPRM3) is a special function register (SFR) which indicates the command parameter. When the host processor sets a parameter to the register for host access (HSTRG: PRMn, n=0 to 9, A to F), the parameter of this register is updated.
Writing to this register from the CPU is disabled.

[Description of Bits]
- **PRMC, PRMD, PRME, PRMF** (31-0)
  Indicates the command parameter from the host processor. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

  The following table shows the symbol name for each parameter register.

Register	Bit[31:24]	Bit[23:16]	Bit[15:8]	Bit[7:0]
HIFPRMF	PRMC[7:0]	PRMD[7:0]	PRME[7:0]	PRMF[7:0]
HIFPRMB	PRM8[7:0]	PRM9[7:0]	PRMA[7:0]	PRMB[7:0]
HIFPRM7	PRM4[7:0]	PRM5[7:0]	PRM6[7:0]	PRM7[7:0]
HIFPRM3	PRM0[7:0]	PRM1[7:0]	PRM2[7:0]	PRM3[7:0]

11.2.11 CPURG: Command Register (HIFCMD)

Address: 0x4005_0030
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	ENT
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CMD1[7:0]								CMD0[7:0]							
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
Command register (HIFCMD) is a read-only special function register (SFR) which indicates the command from the host processor. When the host processor sets a command to the register for host access (HSTRG: CMD*n*), the command of this register is updated. Writing from CPU is invalid.

[Description of Bits]

- **CMD1,0** (bit 15-0)
  Indicates the command from the host processor. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

- **ENT** (bits 16)
  Indicates whether the interrupt request from the host processor exists or not. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".
  When the host processor sets this bit to "1", the CPU is notified of the interrupt, and this bit is updated. When the host processor sets "0" to this bit, the interrupt is not notified to the CPU.
  The value that has been set to this bit is cleared when the CPU writes "1".

ENT	Description
0	Interrupt request from the host processor does not exist (initial value)
1	Interrupt request from the host processor exists

11.2.12 CPURG: Result Register *n* (HIFRLT*n*) *n*: 00 to 3F

Address: 0x4005_0040 to 0x4005_007F
Access: R/W
Access size: 8/16/32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	RSLT03[7:0]								RSLT02[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	RSLT01[7:0]								RSLT00[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write "0" when writing.

[Description of Register]
HIFRLT*n* is a read/write special function register (SFR) that notifies the host processor of the result of command processing.
If a data is set to this register, it is transferred to the register for host access register (HSTRG: RSLT*n*), which can be read by the host processor.

[Description of Bits]
- **RSLT***n* (bit 31-0) *n*:00-3F
  Indicates the result of processing to the host processor. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

  The following table shows the symbol name for each result register.

Register	Bit[31:24]	Bit[23:16]	Bit[15:8]	Bit[7:0]
HIFRLT00W	RSLT03[7:0]	RSLT02[7:0]	RSLT01[7:0]	RSLT00[7:0]
HIFRLT04W	RSLT07[7:0]	RSLT06[7:0]	RSLT05[7:0]	RSLT04[7:0]
HIFRLT08W	RSLT0B[7:0]	RSLT0A[7:0]	RSLT09[7:0]	RSLT08[7:0]
HIFRLT0CW	RSLT0F[7:0]	RSLT0E[7:0]	RSLT0D[7:0]	RSLT0C[7:0]
HIFRLT10W	RSLT13[7:0]	RSLT12[7:0]	RSLT11[7:0]	RSLT10[7:0]
HIFRLT14W	RSLT17[7:0]	RSLT16[7:0]	RSLT15[7:0]	RSLT14[7:0]
HIFRLT18W	RSLT1B[7:0]	RSLT1A[7:0]	RSLT19[7:0]	RSLT18[7:0]
HIFRLT1CW	RSLT1F[7:0]	RSLT1E[7:0]	RSLT1D[7:0]	RSLT1C[7:0]
HIFRLT20W	RSLT23[7:0]	RSLT22[7:0]	RSLT21[7:0]	RSLT20[7:0]
HIFRLT24W	RSLT27[7:0]	RSLT26[7:0]	RSLT25[7:0]	RSLT24[7:0]
HIFRLT28W	RSLT2B[7:0]	RSLT2A[7:0]	RSLT29[7:0]	RSLT28[7:0]
HIFRLT2CW	RSLT2F[7:0]	RSLT2E[7:0]	RSLT2D[7:0]	RSLT2C[7:0]
HIFRLT30W	RSLT33[7:0]	RSLT32[7:0]	RSLT31[7:0]	RSLT30[7:0]
HIFRLT34W	RSLT37[7:0]	RSLT36[7:0]	RSLT35[7:0]	RSLT34[7:0]
HIFRLT38W	RSLT3B[7:0]	RSLT3A[7:0]	RSLT39[7:0]	RSLT38[7:0]
HIFRLT3CW	RSLT3F[7:0]	RSLT3E[7:0]	RSLT3D[7:0]	RSLT3C[7:0]

11.2.13 HSTRG: Configuration Register (CFG)

Address: 0x00 (write), 0x80 (read)
Access: R/W
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	REGMD	-	INTPW[1:0]		INT1EN	INTLVL	-	-
Access	R/W	–	R/W	R/W	R/W	R/W	–	–
Initial value	0	0	0	0	0	0	0	0

CFG is a register to set the configuration of the host interface. It can be read/written from the host processor.

Description of Bits

- **INTLVL** (bit 2)
  This bit controls whether the interrupt signal to the host processor is a level output or pulse output.
  This bit is common to INT0_S and INT1_S.
  For pulse output, the width set in INTPW is used as the pulse width.

INTLVL	Description
0	Pulse output (initial value)
1	Level output

- **INT1EN** (bit 3)
  Controls the INT1_S interrupt signal.

INT1EN	Description
0	INT1_S pin is merged with INT0_S to be output (initial value)
1	INT1_S pin is enabled

- **INTPW** (bit 5-4)
  This bit indicates the pulse width setting when the interrupt signal is a pulse signal.
  If the pulse width is set to 500[ns] or longer, an interrupt pulse may not be output depending on the timing when the CPU writes to the interrupt request register.
  In this case, use the level output as the interrupt signal instead of the pulse output.

INTPW[1:0]	Description
00	250[ns] (4 MHz cycle) (initial value)
01	500[ns] (2 MHz cycle)
10	1000[ns] (1 MHz cycle)
11	2000[ns] (500 kHz cycle)

- **REGMD** (bit 7)
  This bit shows the register access mode of the serial interface (SPI/I2C). When set to "0", the internal address is incremented by 1 each time a 1-byte data is transmitted/received. When set to "1", the address is fixed to the same address.

REGMD	Description
0	Address increment enabled (initial value)
1	Address increment disabled

FIFO register (FIFO) and result register 3F(RSLT3F) are excluded from the address increment. If FIFO register or result register 3F is accessed with this bit set to "0", the address increment is not executed.

11.2.14 HSTRG: Interrupt Mask Register 0 (INTMSK0)

Address: 0x02 (write), 0x82 (read)
Access: R/W
Access size: 8 Bits
Initial Value: 0xFF

Bit	7	6	5	4	3	2	1	0
Symbol name	INT MSK0[7]	INT MSK0[6]	INT MSK0[5]	INT MSK0[4]	INT MSK0[3]	INT MSK0[2]	INT MSK0[1]	INT MSK0[0]
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	1	1	1	1	1	1	1	1

INTMSK0 is a register that masks the interrupt signal to the host processor that corresponds to the interrupt source. It can be read/written from the host processor.

Description of Bits

- **INTMSK0[7:0]** (bit 7-0)
  Masks interrupt sources due to the interrupt request register.

INTMSK0[n]	Description
0	Does not mask the interrupt source due to the HSTRG:INTREQ0 REQ0[n] bit
1	Masks the interrupt source due to the HSTRG:INTREQ0 REQ0[n] bit (initial value)

n=0 to 7

For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.15 HSTRG: Interrupt Mask Register 1 (INTMSK1)

Address: 0x03 (write), 0x83 (read)
Access: R/W
Access size: 8 Bits
Initial Value: 0xFF

Bit	7	6	5	4	3	2	1	0
Symbol name	INT MSK1[7]	INT MSK1[6]	INT MSK1[5]	INT MSK1[4]	INT MSK1[3]	INT MSK1[2]	INT MSK1[1]	INT MSK1[0]
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	1	1	1	1	1	1	1	1

INTMSK1 is a register that masks the interrupt signal to the host processor that corresponds to the interrupt source. It can be read/written from the host processor.

Description of Bits

- **INTMSK1[7:0]** (bit 7-0)
  Masks interrupt sources due to the interrupt request register.

INTMSK1[n]	Description
0	Does not mask the interrupt source due to the HSTRG:INTREQ1 REQ1[n] bit
1	Masks the interrupt source due to the HSTRG:INTREQ1 REQ1[n] bit (initial value)

n=0 to 7

For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.16 HSTRG: Operation Status Register (STATUS)

Address: 0x89 (read)
Access: R
Access size: 8 Bits
Initial Value: 0xFE

Bit	7	6	5	4	3	2	1	0
Symbol name	ST[7]	ST[6]	ST[5]	ST[4]	ST[3]	ST[2]	ST[1]	ST[0]
Access	R/-	R/-	R/-	R/-	R/-	R/-	R/-	R/-
Initial value	1	1	1	1	1	1	1	0

STATUS is a register which indicates the measurement status. It can be read from the host processor. Write to this bit is ignored.

Description of Bits

- **ST[7:0]** (bit 7-0)

  Indicates the measurement status. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.17 HSTRG: Error Code Register 0 (ERROR0)

Address: 0x8A (read)
Access: R
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	ER0[7]	ER0[6]	ER0[5]	ER0[4]	ER0[3]	ER0[2]	ER0[1]	ER0[0]
Access	R/-	R/-	R/-	R/-	R/-	R/-	R/-	R/-
Initial value	0	0	0	0	0	0	0	0

ERROR0 is a read-only register that shows the error code. Writing is invalid.

Description of Bits

- **ER0[7:0]** (bit 7-0)

  Represents an error code. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.18 HSTRG: Error Code Register 1 (ERROR1)

Address: 0x8B (read)
Access: R
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	ER1[7]	ER1[6]	ER1[5]	ER1[4]	ER1[3]	ER1[2]	ER1[1]	ER1[0]
Access	R/-	R/-	R/-	R/-	R/-	R/-	R/-	R/-
Initial value	0	0	0	0	0	0	0	0

ERROR1 is a read-only register that shows the error code. Writing is invalid.

Description of Bits

- **ER1[7:0]** (bit 7-0)

  Sets the interrupt source of the error code to the host processor. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.19 HSTRG: Interrupt Request Register 0 (INTREQ0)

Address: 0x8C (read)
Access: R
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	REQ0 [7]	REQ0 [6]	REQ0 [5]	REQ0 [4]	REQ0 [3]	REQ0 [2]	REQ0 [1]	REQ0 [0]
Access	R/-	R/-	R/-	R/-	R/-	R/-	R/-	R/-
Initial value	0	0	0	0	0	0	0	0

INTREQ0 is a read-only register that shows the interrupt source. Each value of this register is cleared by being read by the host processor. Writing is invalid.

Description of Bits

- **REQ0[7:0]** (bit 7-0)
  Indicates the source of interrupt to the host processor.
  The second overflow of WDT results in REQ0[7:0]=0xFF.

REQ0[n]	Description
0	No interrupt source (initial value)
1	Interrupt source exists

n=0 to 7

For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.20 HSTRG: Interrupt Request Register 1 (INTREQ1)

Address: 0x8D (read)
Access: R
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	REQ1 [7]	REQ1 [6]	REQ1 [5]	REQ1 [4]	REQ1 [3]	REQ1 [2]	REQ1 [1]	REQ1 [0]
Access	R/-	R/-	R/-	R/-	R/-	R/-	R/-	R/-
Initial value	0	0	0	0	0	0	0	0

INTREQ1 is a read-only register that shows the interrupt source. Each value of this register is cleared by being read by the host processor. Writing is invalid.

Description of Bits

- **REQ1[7:0]** (bit 7-0)
  Indicates the source of interrupt to the host processor.
  The second overflow of WDT results in REQ1[7:0]=0xFF.

REQ1[n]	Description
0	No interrupt source (initial value)
1	Interrupt source exists

n=0 to 7

For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".
The timing of clearing INTREQ1,0 depends on the INT1EN bit setting of the configuration register (HSTRG:CFG).
For details, see "11.4.5Timing of Clearing Interrupt Request Register".

11.2.21 HSTRG: FIFO Register (FIFO)

Address: 0x10 (write), 0x90 (read)
Access: R/W
Access size: 8 Bits
Initial Value: 0xXX

Bit	7	6	5	4	3	2	1	0
Symbol name	FIFO [7]	FIFO [6]	FIFO [5]	FIFO [4]	FIFO [3]	FIFO [2]	FIFO [1]	FIFO [0]
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	x	x	x	x	x	x	x	x

FIFO register is a register that can be written/read from the host processor.
This register can be written/read from the host processor only when FSEL of the CPU register is set to 1. When FSEL is set to 0, writing is disabled and reading gets Undefined.

Description of Bits

- **FIFO[7:0]** (bit 7-0)

  Indicates the FIFO data. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.22 HSTRG: Parameter Register *n* (PRM*n*) *n*:00 to 0F

Address: 0x20 to 0x2F (write), 0xA0 to 0xAF (read)
Access: R/W
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	PRM*n* [7]	PRM*n* [6]	PRM*n* [5]	PRM*n* [4]	PRM*n* [3]	PRM*n* [2]	PRM*n* [1]	PRM*n* [0]
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

PRM*n* is a register that can be written/read from the host processor that sets the command parameter.

Description of Bits

- **PRM*n*[7:0]** (bit 7-0)

  Represents a command parameter. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.23 HSTRG: Command Register 0*n* (CMD*n*) *n*:0 to 1

Address: 0x30, 0x31 (write), 0xB0, 0xB1 (read)
Access: R/W
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	CMD*n* [7]	CMD*n* [6]	CMD*n* [5]	CMD*n* [4]	CMD*n* [3]	CMD*n* [2]	CMD*n* [1]	CMD*n* [0]
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

CMD*n* is a register that can be written/read from the host processor that sets the command.

Description of Bits

- **CMD*n*[7:0]** (bit 7-0)

  Represents a command. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.2.24 HSTRG: Command Entry Register (ENT)

Address: 0x32 (write), 0xB2 (read)
Access: R/W
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	ENT
Access	–	–	–	–	–	–	–	R/W
Initial value	0	0	0	0	0	0	0	0

ENT is a register to request an interrupt for CPU. When this register is set to "1", an interrupt to CPU is generated.
The value that has been set to this register is cleared when the CPU writes "1" to the bit 16 of the command register (CPURG:HIFCMD).

Description of Bits

- **ENT** (bit 0)
  Indicates whether the interrupt request from the host processor exists or not.

ENT	Description
0	Interrupt request from the host processor does not exist (initial value)
1	Interrupt request from the host processor exists

11.2.25 HSTRG: Result Register *n* (RSLT*n*) *n*: 00 to 3F

Address: 0xC0 to 0xFF (read)
Access: R
Access size: 8 Bits
Initial value: 0x00

Bit	7	6	5	4	3	2	1	0
Symbol name	RSLT*n* [7]	RSLT*n* [6]	RSLT*n* [5]	RSLT*n* [4]	RSLT*n* [3]	RSLT*n* [2]	RSLT*n* [1]	RSLT*n* [0]
Access	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0

RSLT*n* is a read-only register that indicates the command processing result. Writing is invalid.

Description of Bits

- **RSLT*n*[7:0]** (bit 7-0)

  Indicates the result of command processing to the host processor. For details, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.3 Serial Interface

I2C is selected during reset. I2C or SPI (three-wire or four-wire) is set by the IFSEL and IFCFG bits of CPURG: configuration register (HIFCFG) after the system reset is released.

11.3.1 I2C Interface

I2C interface can be used for communication with the host processor by setting the IFSEL bit of GPURG: configuration register to "1".

The following shows an I2C connection example. When the interrupt signals (INT0_S, INT1_S) are not used, up to four IO ports can be used for other uses.

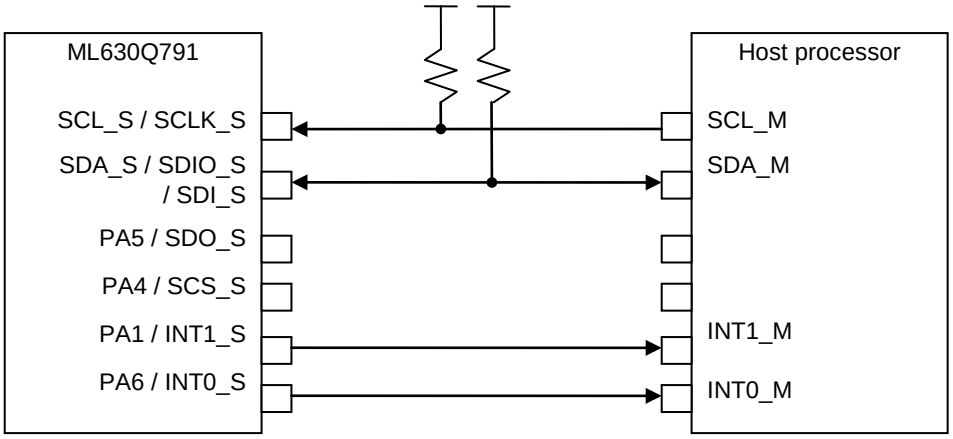


Figure 11-2 I2C Connection Example

11.3.1.1 I2C Slave Address

The slave address of I2C can be set by IFCFG[7:1] of CPURG: configuration register as shown in Figure 11-3. The initial value is 0x17. (address for write: 0x2E, address for read: 0x2F)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Slave address	IFCFG[7] 0	IFCFG[6] 0	IFCFG[5] 1	IFCFG[4] 0	IFCFG[3] 1	IFCFG[2] 1	IFCFG[1] 1	R/W

Figure 11-3 I2C Slave Address Format

11.3.1.2 I2C Transfer Format

When writing from the master to slave, sequential writing in the order of "internal address, internal write data, internal write data" is possible as shown in Figure 11-4.

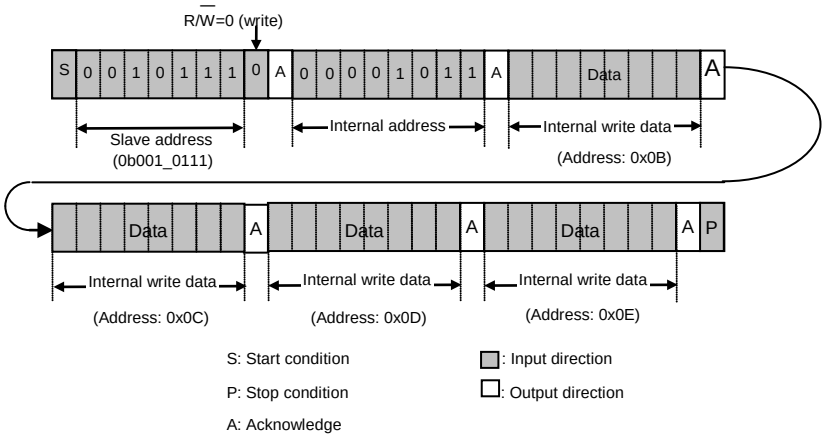


Figure 11-4 Register Write Transfer Format

When reading the slave data from the master, write access is required in order to set the internal read address. This address is maintained until updated.

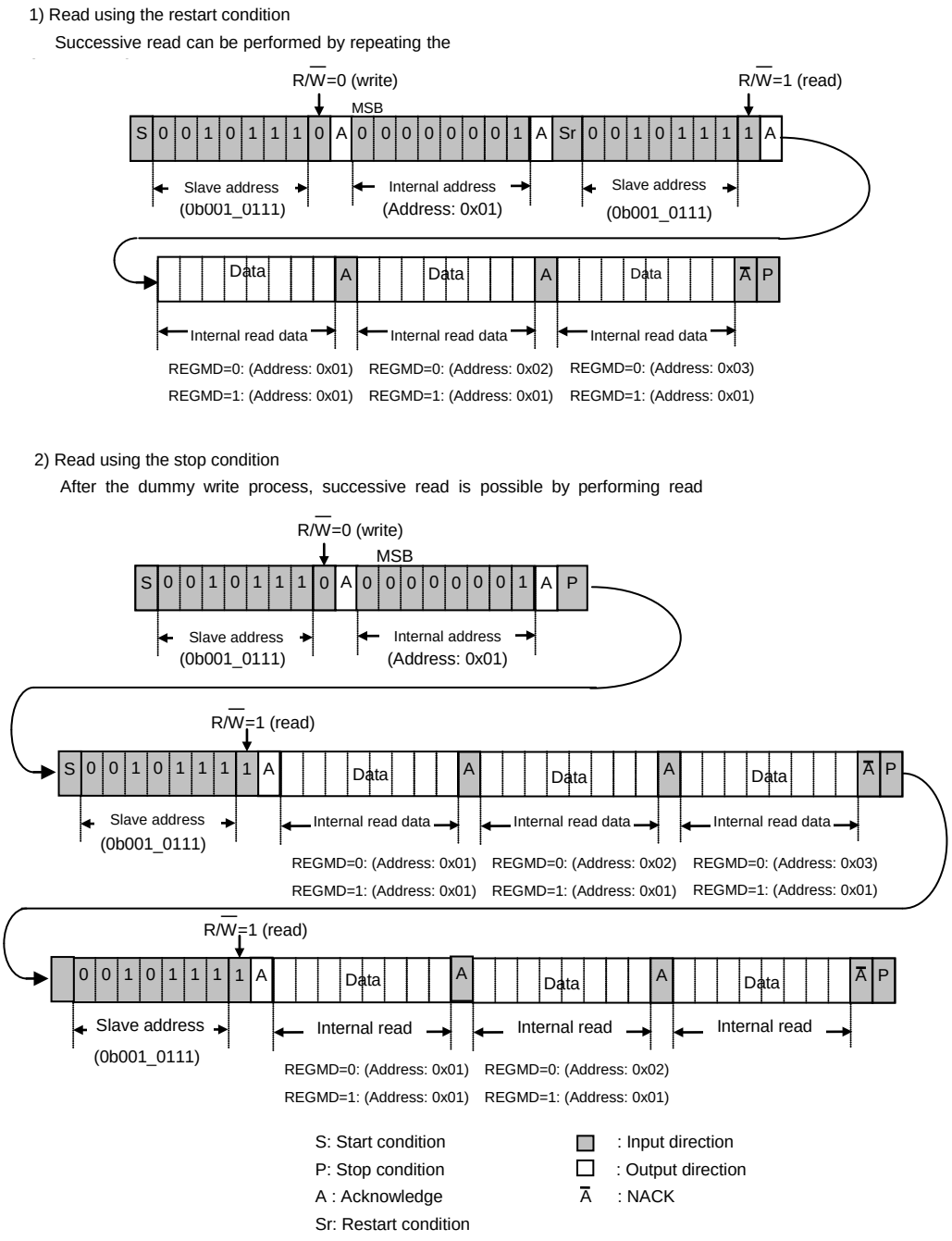


Figure 11-5 Register Read Transfer Format

11.3.2 SPI Interface

SPI interface can be used for communication with the host processor by setting the IFSEL bit of GPURG: configuration register to "0".

Three-wire or four-wire can be selected by IFCFG[7]. Figure 11-6 and Figure 11-7 show examples of connection with the host processor. IFCFG[6] sets Hi-Z or 0 output for data output in the SPI no output mode. IFCFG[5] sets the polarity of the chip select signal SCS_S.

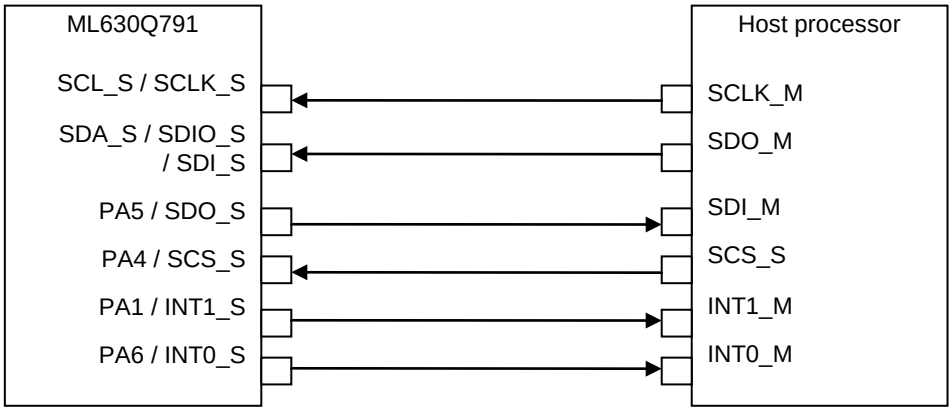


Figure 11-6 SPI (Four-Wire) Connection Example

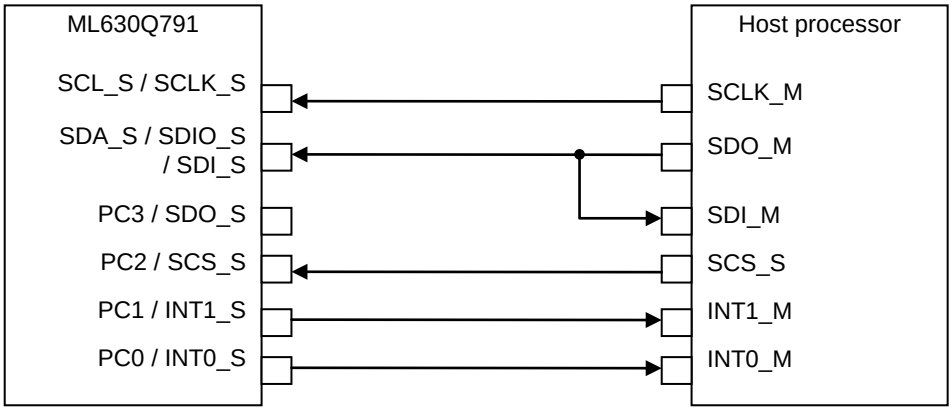


Figure 11-7 SPI (Three-Wire) Connection Example

11.3.2.1 SPI Transfer Format

Figure 11-8 and Figure 11-9 show the transfer format of the write sequence.
Figure 11-10 and Figure 11-11 show the transfer format of the read sequence.

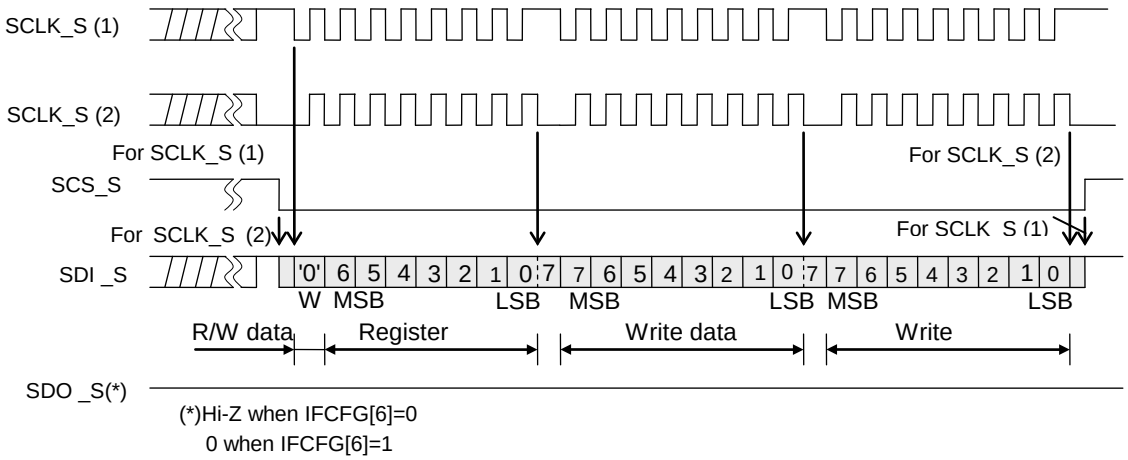


Figure 11-8 Write Sequence (Four-Wire)

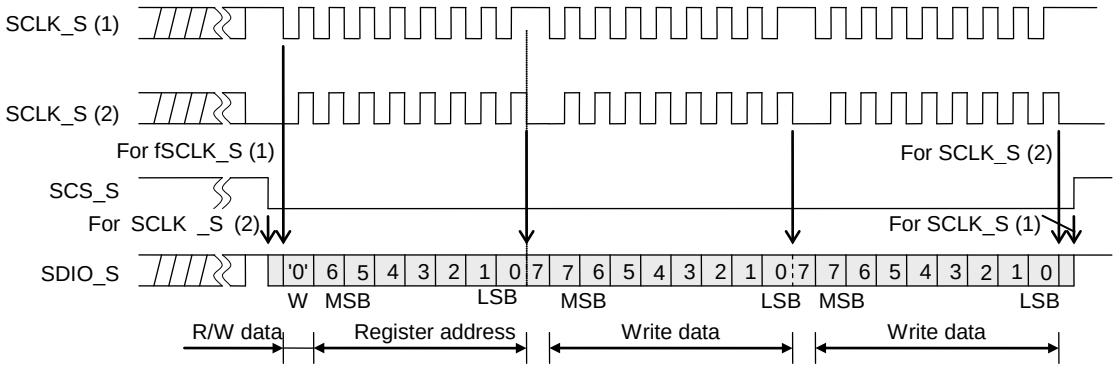

SCLK_S (1)
SCLK_S (2)
For fSCLK_S (1)
For SCLK_S (2)
SCS_S
For SCLK _S (2)
For SCLK_S (1)
SDIO_S
'0' 6 5 4 3 2 1 0 7 7 6 5 4 3 2 1 0 7 7 6 5 4 3 2 1 0
W MSB LSB MSB LSB MSB LSB
R/W data
Register address
Write data
Write data

Figure 11-9 Write Sequence (Three-Wire)

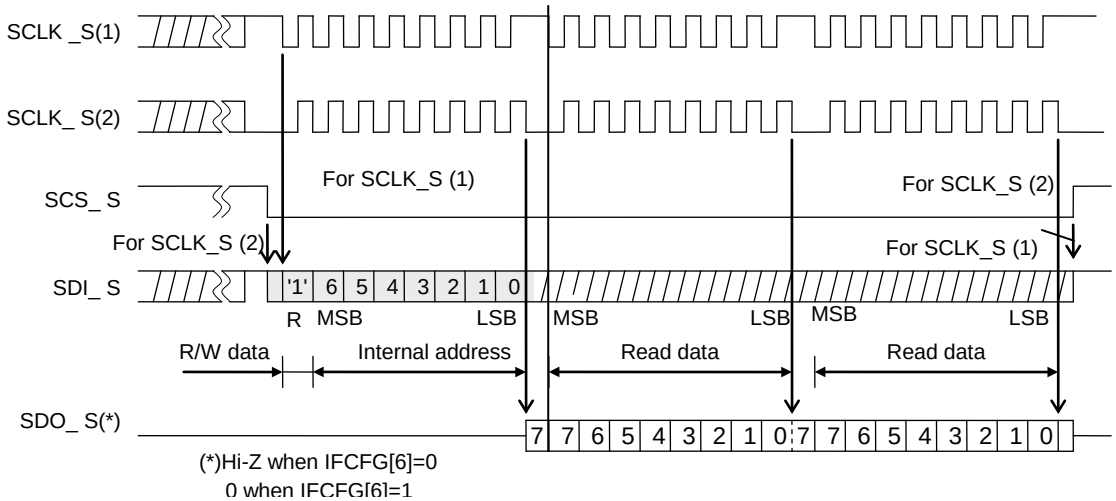


Figure 11-10 Read Sequence (Four-Wire)

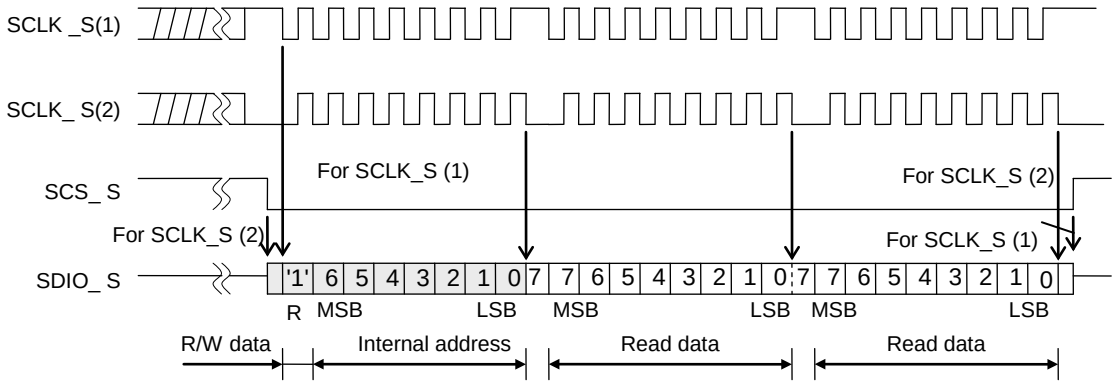

SCLK _S(1)
SCLK_ S(2)
SCS_ S
For SCLK_S (1)
For SCLK_S (2)
For SCLK_S (2)
For SCLK_S (1)
SDIO_ S
'1' 6 5 4 3 2 1 0 7 7 6 5 4 3 2 1 0 7 7 6 5 4 3 2 1 0
R MSB LSB MSB LSB MSB LSB
R/W data
Internal address
Read data
Read data

Figure 11-11 Read Sequence (Three-Wire)

11.4 Description of Operation

11.4.1 Power-down and Resume

This LSI can conserve the power consumption by transitioning to the sleep mode while waiting for a command.

Resume from the sleep mode is possible by entering an external interrupt or writing "1" to the bit0 (ENT) of HSTRG command entry register (0x32) from the host processor.

Note on the power-down is as follows.
Confirm that both the RLTP and INTP bits of the operation status register (HIFST) are set to "0" before power-down.

11.4.2 Command Input and Processing of Response

The host processor sets a parameter to the HSTRG parameter register 00 to 0F (0x20 to 0x2F) and a command to the HSTRG command register 0 and 1 (0x30 and 0x31). Then, writing "1" to the bit 0 (ENT) of the HSTRG command entry register (0x32) causes an interrupt to the CPU for the command processing.

After the CPU accepts the command, and "1" is written to the bit 16 (ENT) of the CPURG command register, the HSTRG command entry register (0xB2) is cleared.

The host processor can set the next parameter and command after confirming that the bit 0 (ENT) of HSTRG command entry register (0xB2) is cleared. If the next command or parameter is set before the CPU accepts the command, malfunction occurs.

The CPU processes the accepted command, sets the result to the CPURG result register 00 to 3F, and then sets the interrupt source to the CPURG interrupt request register 0 and 1. At this time, an interrupt can be output to the host processor if no interrupt mask is set to the HSTRG interrupt mask registers 0 and 1 (0x82 and 0x83).

The operation of interrupt signal differs depending on the INT1EN bit of configuration register (CPURG:HIFCFG) as follows.

- For INT1EN=0
  ORed sources of the interrupt request register 0 and 1 are output from INT0_S.
- For INT1EN=1
  The sources of the interrupt request register 0 and 1 are output from INT0_S and INT1_S respectively.

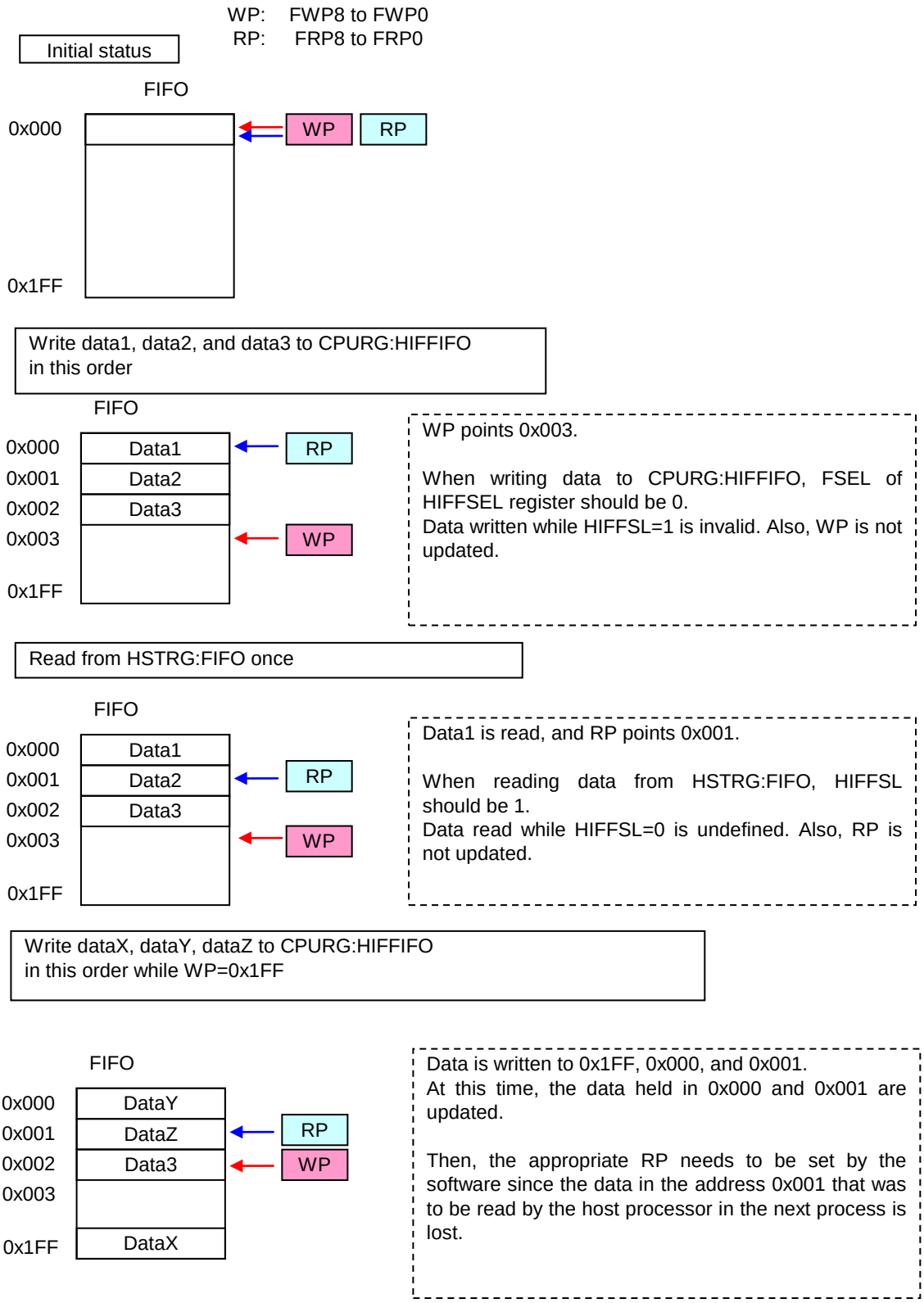
When the host processor receives the interrupt, it reads HSTRG interrupt request register 0 and 1 (0x82 and 0x83). Then, it reads the data that has been written to the CPURG result register, from the HSTRG result register 00 to 3F (0xC0 to 0xFF).

The interrupt to the host processor is output in the negative logic. For setting and clearing the interrupt, see "11.2.5 CPURG: Interrupt Request Register (HIFRQ)".

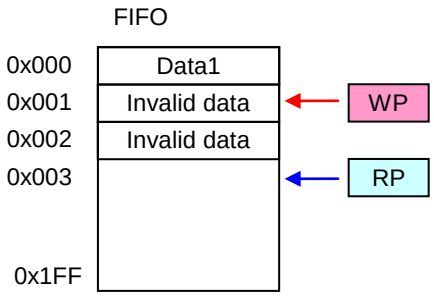
For commands and responses of results, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.4.3 Write/Read Data to/from FIFO

Access to CPURG:HIFFIFO or HSTRG:FIFO means writing/reading data to/from 512B FIFO.
An example is shown below.
The CPU and the host write/read in the units of one byte.



Write data1 to the CPURG:HIFFIFO from the initial state, and then read from HSTRG:FIFO three times



Data1 is read.

If data is read beyond WP, the data is invalid.
Control by software is required so that RP does not exceed WP.

11.4.4 Register Access Conflict by Host and CPU

Only the interrupt request register and the result register among the host interface registers can be accessed without handshake at conflict. The other registers should be controlled to prevent access conflicts by handshake between the host and the CPU.

[Notes]
For FIFO register access, the FIFO switch register (CPURG:HIFFSEL) should be controlled.

The interrupt request register and result register are controlled as follows.

-Interrupt request register 0,1
If a conflict occurs on this register, writing from the CPU is executed after clearing by reading from the host. The INTP bit of the CPUG: operation status register is set to "1" while writing from the CPU is held.

-Result register 00-3F
The host can read data by write data unit of the CPU (2 byte or 4 byte) while keeping a consistency.
However, it is only true with the host in the address increment mode, successive read, and data in the 4-byte boundary.

11.4.5 Timing of Clearing Interrupt Request Register

Timing of clearing the interrupt request register depends on the INT1EN bit setting of the configuration register (HSTRG:CFG), as described below.

- When INT1EN="0"
  INT0_S is the logical ORed output of REQ0 and REQ1.
  REQ0 and 1 are both cleared when REQ0 is read.
  At this time, REQ1 is saved to the buffer for next transmission.
  If only REQ1 is read from the host, REQ1 is not cleared. Be sure to read REQ0 and 1 successively.

- When INT1EN="1"
  REQ0 is cleared when REQ0 is read, and REQ1 is cleared when REQ1 is read.

11.4.6 Error Handling

Should an error occurs while processing a command or during any other operation, the CPU sets an error code to the CPURG operation status registers ER0 and ER1. The host processor can read the error code that has been set from the HSTRG error code registers (0x8A and 0x8B). Depending on the contents of the error that has been set, the host processor needs to handle the situation.

For the details of error codes, see "ML630Q791 User's Guide SDK Sensor Control Host IF Specification ".

11.4.7 Error Status Notification

If the WDTP of WDT is not cleared and WDT overflow occurs, an interrupt for the host processor is output to notify that this LSI is in the abnormal state. When this LSI in the abnormal state, only the value 0xFF can be read from the HSTRG interrupt request registers 0 and 1 (0x8C and 0x8D). The host processor determines whether this LSI is in an abnormal state or not through the values of HSTRG interrupt request registers 0 and 1 (0x8C and 0x8D). If determined as an abnormal state, please perform the hardware reset.

11.4.8 Clock Requirements

The system clock should be operated at 32[MHz] to use the host IF properly.

11.5 Specifying Port Registers

When using the SPI interface of the host interface or interrupt (INT0_S or INT1_S), related port register bits need to be set. See Chapter 15, "GPIO" for details about the port registers.

11.5.1 When Using SPI Interface (Three-wire)

Set PA4 (bit 17, 16) of the PAMOD register to "01" to select the secondary function (SCS_S) of PA4.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**0**	**1**

* : This bit is not related to the SPI interface function

[Note]
PIODIR and PIOCON registers do not need to be set. When the secondary function is selected, the setting is automatically changed, but the value of each register does not change.

11.5.2 When Using SPI Interface (Four-wire)

Set PA4 (bit 17, 16) of the PAMOD register to "01" to select the secondary function (SCS_S) of PA4.
Set PA5 (bit 21, 20) of the PAMOD register to "01" to select the secondary function (SDO_S) of PA5.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	**0**	**1**	*	*	**0**	**1**

Set the bit 5 of the PIODIR register to "0" to set the input/output mode of PA5 to output.
When the secondary function is selected, the setting of the bit 4 is automatically changed, but the value of the bit does not change.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR[6:0]						
Setting value	*	*	*	*	*	*	*	*	*	*	**0**	*	*	*	*	*

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : This bit is not related to the SPI interface function

[Note]
The PIOCON register does not need to be set. When the secondary function is selected, the setting is automatically changed, but the value of the PIOCON register does not change.

11.5.3 When INT0_S Is Used

Set PA6 (bit 25 and 24) of the PAMOD register to "01" to select the secondary function of PA6.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	**0**	**1**	*	*	*	*	*	*	*	*

Set the bit 6 of the PIODIR register to "0" to set the input/output mode of PA6 to output.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR[6:0]						
Setting value	*	*	*	*	*	*	*	*	*	**0**	*	*	*	*	*	*

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the INT0_S function

[Note]
The PIOCON register does not need to be set. When the secondary function is selected, the setting is automatically changed, but the value of the PIOCON register does not change.

11.5.4 When INT1_S Is Used

Set PA1 (bit 5 and 4) of the PAMOD register to "10" to select the tertiary function of PA1.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	**1**	**0**	*	*	*	*

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Set the bit 1 of the PIODIR register to "0" to set the input/output mode of PA1 to output.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR[6:0]						
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**0**	*

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the INT1_S function

[Note]
The PIOCON register does not need to be set. When the tertiary function is selected, the setting is automatically changed, but the value of the PIOCON register does not change.

Chapter 12 I2C Bus Interface

12. I^2C Bus Interface

12.1 Overview

This LSI includes two channels of I^2C bus interface (master) that conforms to the typical I^2C bus specification.

12.1.1 Features

- The master function is supported.
- The communication speed can be set based on the system clock.
  (Only a 32 MHz system clock is supported.)
- The 32-byte buffer function is provided. Whether or not to use the buffer mode can be selected.

12.1.2 Configuration

Figure 12-1 shows the configuration diagram of the I^2C.

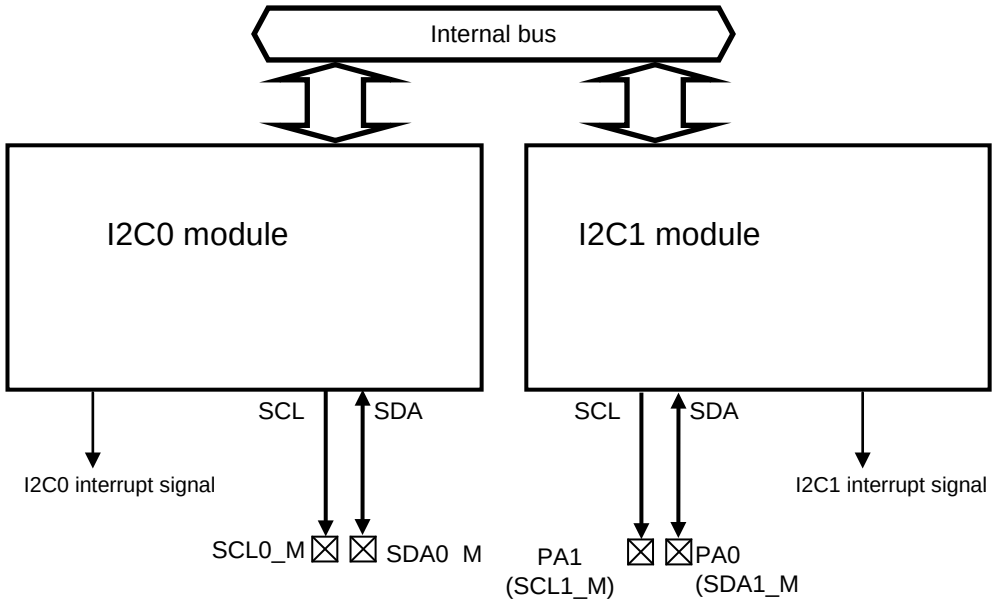


Figure 12-1 Configuration Diagram

12.1.3 List of Pins

Table 12-11 List of Pins Interfaced with the Outside of LSI

Pin name	I/O	Initial status	Initial value	Description
SDA0_M	I/O	I	-	Serial data input/output pin
SCL0_M	O	I	-	Serial data transfer clock
PA0 (SDA1_M)	I/O	I	-	Serial data input/output pin
PA1 (SCL1_M)	O	I	-	Serial data transfer clock

[Note]
The initial statuses/initial values of SDA1_M and SCL1_M are the values when the secondary functions of the corresponding pins are selected. Therefore, they are different from the initial values of the LSI pins.
For details of switching to the secondary functions of the LSI pins, see the description of port A mode setting register in Chapter 5 "MCU Control Function".

12.2 Description of Registers

12.2.1 List of Registers

Table 12-2 List of Registers

Channel	Address	Name	Symbol	R/W	Size [bits]	Initial value
ch0	0x4008_3004	I^2C control register 0	I2CCTL0	R/W	32	0x0000_0000
	0x4008_3008	I^2C status register 0	I2CSR0	R/W	32	0x0000_0000
	0x4008_300C	I^2C data register 0	I2CDR0	R/W	32	0x0000_0000
	0x4008_3010	I^2C bus monitor register 0	I2CMON0	R	32	0x0000_0003
	0x4008_3014	I^2C bus transfer rate setup counter 0	I2CBC0	R/W	32	0x0000_0000
	0x4008_3018	I^2C mode register 0	I2CMOD0	R/W	32	0x0000_0000
	0x4008_301C	I^2C buffer mode slave address register 0	I2CBUFSLV0	R/W	32	0x0000_0000
	0x4008_3020	I^2C buffer mode sub address register 0	I2CBUFSUB0	R/W	32	0x0000_0000
	0x4008_3024	I^2C buffer mode format register 0	I2CBUFFOR0	R/W	32	0x0000_0000
	0x4008_3028	I^2C buffer mode control register 0	I2CBUFCTL0	R/W	32	0x0000_0000
	0x4008_302C	I^2C buffer mode interrupt mask register 0	I2CBUFMSK0	R/W	32	0x0000_0000
	0x4008_3030	I^2C buffer mode status register 0	I2CBUFSTA0	R/W	32	0x0000_0000
	0x4008_3034	I^2C buffer mode level register 0	I2CBUFLEV0	R/W	32	0x0000_0000
	0x4008_3048	I^2C timer register 0	I2CTMR0	R/W	32	0x0000_0000
	0x4008_3050	I^2C input noise filter setting register 0	I2CNF0	R/W	32	0x0000_0001
ch1	0x4008_3404	I^2C control register 1	I2CCTL1	R/W	32	0x0000_0000
	0x4008_3408	I^2C status register 1	I2CSR1	R/W	32	0x0000_0000
	0x4008_340C	I^2C data register 1	I2CDR1	R/W	32	0x0000_0000
	0x4008_3410	I^2C bus monitor register 1	I2CMON1	R	32	0x0000_0003
	0x4008_3414	I^2C bus transfer rate setup counter 1	I2CBC1	R/W	32	0x0000_0000
	0x4008_3418	I^2C mode register 1	I2CMOD1	R/W	32	0x0000_0000
	0x4008_341C	I^2C buffer mode slave address register 1	I2CBUFSLV1	R/W	32	0x0000_0000
	0x4008_3420	I^2C buffer mode sub address register 1	I2CBUFSUB1	R/W	32	0x0000_0000
	0x4008_3424	I^2C buffer mode format register 1	I2CBUFFOR1	R/W	32	0x0000_0000
	0x4008_3428	I^2C buffer mode control register 1	I2CBUFCTL1	R/W	32	0x0000_0000
	0x4008_342C	I^2C buffer mode interrupt mask register 1	I2CBUFMSK1	R/W	32	0x0000_0000
	0x4008_3430	I^2C buffer mode status register 1	I2CBUFSTA1	R/W	32	0x0000_0000
	0x4008_3434	I^2C buffer mode level register 1	I2CBUFLEV1	R/W	32	0x0000_0000
	0x4008_3448	I^2C timer register 1	I2CTMR1	R/W	32	0x0000_0000
	0x4008_3450	I^2C input noise filter setting register 1	I2CNF1	R/W	32	0x0000_0001

12.2.2 I²C Control Register (I2CCTL0, 1)

Address: 0x4008_3004 (0ch), 0x4008_3404 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	I2CCS	I2CDR_LEN	I2CSTPIE	-*	I2CCFIE	I2CALIE	I2CMEN	I2CAASIE	I2CMSTA	I2CMTX	I2CTXAK	I2CRSTA	I2CMD[1:0]	
Access	-	-	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register controls the transmission and reception of the I²C bus. This register can be read from or written to by software. It controls only the interrupts of the interrupt enable registers, namely I2CAASIE, I2CALIE, I2CCFIE, I2CSTPIE, I2CDR_LDIE, I2CISTPIE and I2CNSTPIE, and does not control the corresponding bits of the status register. (The status register will change even if an interrupt is disabled.)
Only the I2CMD and I2CMEN bits are used in the buffer mode.

[Description of Bits]

- **I2CMD[1:0]** (bits 0-1)
  Select the standard-mode or the fast-mode.

I2CMD[1:0]	Description
00	Selects the standard-mode (100 kHz).
01	Selects the fast-mode (400 kHz).
10	Setting prohibited
11	Setting prohibited

- **I2CRSTA** (bit 2)
  Specifies to transmit the repeated START conditions. If “1” is written to this bit when the I²C module is a bus master, a repeated START condition is sent to the bus. This bit is automatically reset to “0” after sending a repeated START condition.

[Note]
When sending a repeated START condition, overwrite I2CMSTA as is with the setting of “1”. Operation cannot be guaranteed if “1” is written to this bit I2CRSTA and “0” to the I2CMSTA bit. Because this bit is automatically reset to “0” after sending the repeated START condition, if another bit in the control register will be set after this bit is set to “1”, set this bit to “0” or keep the previous value. If “1” is written again, the repeated START condition will be sent at that very moment. This bit also indicates the status of ROM controller. This bit is “1” during sector erase/1-word write. It automatically changes to “0” when sector erase/1-word write is completed.

- **I2CTXAK** (bit 3)
  Specifies transmission of ACK or NACK in the receive mode. The acknowledge data that was set to this bit in advance is sent to the transmit device after data is received.

I2CTXAK	Description
0	At acknowledge output timing, outputs “0”. ACK output
1	At acknowledge output timing, outputs “1”. NACK output

- **I2CMTX** (bit 4)
  Selects the data transfer direction.

I2CMTX	Description
0	Received by master.
1	Transmitted by master.

- **I2CMSTA** (bit 5)
  Specifies to transmit a START condition or a STOP condition. If this bit is rewritten from “0” to “1”, a start sequence is sent to the bus. When this bit is cleared, a stop sequence is sent.

I2CMSTA	Description
0	Sends a STOP condition.
1	Sends a START condition.

- **I2CAASIE** (bit 6)
  Specifies to enable or disable an MAAS interrupt. The I2CMAAS status will change even if an interrupt is disabled by this bit.

I2CAASIE	Description
0	Disables MAAS interrupt.
1	Enables MAAS interrupt.

- **I2CMEN** (bit 7)
  Initializes this I^2C module. Set this bit to “1” when using this I^2C module.

I2CMEN	Description
0	Initializes this I^2C module
1	Enables this I^2C module.

[Note]
If the I2CMEN bit is set to “0”, the I^2C bus control section, I^2C status register (including the I2CMBB bit), I^2C buffer level register, and I^2C buffer mode status register will be initialized.
The I^2C buffer mode format register, I^2C control register, I^2C data register, I^2C bus monitor register, I^2C bus transfer rate setup counter, I^2C mode register, I^2C buffer mode slave address register, I^2C buffer mode sub address register, I^2C buffer mode interrupt mask register, and I^2C timer register will not be initialized.

- **I2CALIE** (bit 8)
  Specifies to enable or disable an MAL interrupt. The I2CMAL status will change even if an interrupt is disabled by this bit.

I2CALIE	Description
0	Disables MAL interrupt.
1	Enables MAL interrupt.

- **I2CCFIE** (bit 9)
  Specifies to enable or disable an MCF interrupt. The I2CMCF status will change even if an interrupt is disabled by this bit.

I2CCFIE	Description
0	Disables MCF interrupt.
1	Enables MCF interrupt.

- **I2CSTPIE** (bit 11)
  Specifies to enable or disable an STP interrupt. The I2CSTP status will change even if an interrupt is disabled by this bit.

I2CSTPIE	Description
0	Disables STP interrupt.
1	Enables STP interrupt.

- **I2CDR_LDIE** (bit 12)
  Specifies to enable or disable a DR_LD interrupt. The I2CDR_LD status will change even if an interrupt is disabled by this bit.

I2CDR_LDIE	Description
0	Disables DR_LD interrupt.
1	Enables DR_LD interrupt.

[Note]
The DR_LD interrupt is a spare function. It is not used in Section 12.3, "Description of Operation". Normally, disable the DR_LD interrupt by setting this bit to “0”.

- **I2CCS** (bit 13)
  Specifies to halt SCL. SCL stops subsequently to an MCF interrupt that is generated after this bit is set to “1”. When inserting a repeated start after executing the transmit mode, set this bit to “1” during a 1-byte data transfer period immediately before the insertion. After the completion of the transfer, set the I2CRSTA bit to “1” and, at the same time, clear this bit. Specifies to enable or disable an STP interrupt. The I2CSTP status will change even if an interrupt is disabled by this bit.

I2CCS	Description
0	Continues SCL output.
1	Stops SCL output upon completion of the next transfer.

12.2.3 I^2C Status Register (I2CSR0, 1)

Address: 0x4008_3008 (0ch), 0x4008_3408 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	I2CISTP	I2CNSTP	-*	-*	I2CAKMON	-*	-*	I2CRBUF	I2CMCF	-*	I2CMBB	-*	I2CDR_LD	-*	I2CMIF	I2CRXAK
Access	R/W	R/W	-	-	R/W	-	-	R/W	R/W	-	-	-	R/W	-	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
The read data of this register is "0".
*: Reserved bit for future expansion. Write “0” when writing.

[Description of Register]
This register indicates the status of the I^2C bus.
The bits other than the I2CRXAK, I2CMIF, I2CSRW, I2CMBB, I2CRBUF, I2CDRSTA and I2CAKMON bits will be cleared by writing “0”. Only the I2CMBB bit is used in the buffer mode.

[Description of Bits]

- **I2CRXAK** (bit 0)
  Indicates the reception status of ACK/NACK. Acknowledge data to be replied by the receiving device in the transmit mode is stored. This bit is updated every time ACK/NACK is received, and this bit holds the acknowledge data received last even after the bus is released. In the transmit mode, check this bit at the time of an MCF interrupt. If a NACK is received, finish the transfer.

I2CRXAK	Description
0	Received an ACK.
1	Received a NACK.

- **I2CMIF** (bit 1)
  Indicates that an interrupt has been requested. This bit is an interrupt line monitoring bit. If interrupt is enabled for MCF interrupt, MAAS interrupt, MAL interrupt, DR_LD interrupt, STP interrupt, NSTP interrupt, and ISTP interrupt, this bit is set to “1” at the same time that the bit (I2CMCF, I2CDR_LD, I2CNSTP, and I2CISTP) for each interrupt source is set to “1”.
  To clear this bit (clear an interrupt line), it is necessary to write "0" to each of the interrupt source bits (I2CMCF, I2CDR_LD, I2CNSTP, and I2CISTP).

I2CMIF	Description
0	No interrupt request
1	Interrupt request

[Note]
Whether or not to enable each interrupt is set by the I2CCTL register. For details, refer to "I2C Control Register (I2CCTL0, 1)".

- **I2CDR_LD** (bit 3)
  This bit indicates that the transmit buffer is emptied and transmit data can be loaded to the I2CDR register. After this bit has been set to "1" in the transmit mode, the data to be sent next can be written into the data register without destroying the previously transmitted data. This bit is cleared by writing "0" by software. This bit is set to "1" when the transfer of 2 bits out of 8-bit transmit data is finished (at a falling edge of SCL). By using this bit (interrupt), transmit data can be written before an MCF interrupt.

I2CDR_LD	Description
0	Data load to the data register is not allowed.
1	Data load to the data register is allowed.

[Note]
This bit is used as a spare function. It is not used in the operation sequence.

- **I2CMBB** (bit 5)
  Indicates the status of the I^2C bus. This bit is set to "1" when a START condition is detected; this bit is reset to "0" when a STOP condition is detected. By reading this bit, it is possible to check whether the bus is currently occupied or released. This bit is set to "1" after a START condition is detected (at a falling edge of SCL after SDA changes from "1" to "0" when SCL is "1"), and is set to "0" after a STOP condition is detected (at a rising edge of SDA when SCL is "1").

I2CMBB	Description
0	The I^2C bus is open.
1	The I^2C bus is occupied.

[Note]
Before starting a master transmit or master receive transfer, read this bit and make sure that the bus is open.

- **I2CMCF** (bit 7)
  Indicates that data transfer has been completed. This bit is set to "1" when the transmission/reception of 1-byte data(*1) is complete.
  This bit is cleared by writing "0" by software. This bit is set to "1" at a rising edge of SCL when a 1-byte transfer is complete and an ACK response is started.
  (*1) The data signifies all of the 1-byte transfer, which includes the transfer of an address immediately after start/repeated start that the master transmits.

I2CMCF	Description
0	Before data transfer start or during data transfer.
1	Data transfer is completed.

[Note]
In the receive mode, clear this bit by writing "0" to it after reading the receive data from the I2CDR register. If any data is left in the receive buffer, this bit cannot be cleared even if "0" is written to it.

- **I2CRBUF** (bit 8)
  This bit indicates the existence of receive buffer data. If this bit is “1”, it indicates that the receive buffer has data, and the I2CMCF bit is also “1”. When the received data is read from the I2CDR register, this bit is cleared to “0”.

I2CRBUF	Description
0	Data absent in receive buffer
1	Data present in receive buffer

[Note]
This bit is used as a spare function. It is not used in the operation sequence.

- **I2CAKMON** (bit 11)
  This bit is used to monitor the transmission or reception status of ACK/NACK. The ACK/NACK bit can be monitored in all modes regardless of transmission and reception. The value is updated at the timing of the ACK/NACK bit.

I2CAKMON	Description
0	Received or transmitted an ACK.
1	Received or transmitted a NACK.

[Note]
This bit is used as a spare function. It is not used in the operation sequence.

- **I2CNSTP** (bit 14)
  This bit is set to "1" when a NACK is received and transfer is terminated in address transmission in the transmit mode or receive mode. This bit is cleared by writing "0" by software. The interrupt by this bit is notified with the same timing as the I2CMCF interrupt when a NACK is received.

I2CNSTP	Description
0	Has not received a NACK.
1	Received a NACK and terminated transfer.

[Note]
This bit is used as a spare function. It is not used in the operation sequence.

- **I2CISTP** (bit 15)
  This bit is set to "1" if a STOP condition has occurred at unexpected timing, and the transfer has ended abnormally. This bit is set to "1" under any of the following conditions:
  •When a STOP condition is detected before the master itself sends a STOP condition while the master is transmitting or receiving
  This bit is cleared by writing "0" by software.

I2CISTP	Description
0	Has not detected an unexpected STOP condition.
1	Detected an unexpected STOP condition.

[Note]
This bit is used as a spare function. It is not used in the operation sequence.

12.2.4 I²C Data Register (I2CDR0, 1)

Address: 0x4008_300C (0ch), 0x4008_340C (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	I2CDA[7:0]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register sets transmit data or stores received data. In the buffer mode, up to 32 bytes of transmit data can be stored in the buffer by writing this register. When receiving data, the received data stored in the buffer can be read by reading this register.

12.2.5 I^2C Bus Monitor Register (I2CMON0, 1)

Address: 0x4008_3010 (0ch), 0x4008_3410 (1ch)
Access: R
Access size: 32 Bits
Initial value: 0x0000_0003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	DBMON3				DBMON2				DBMON1				-*	-*	I2CSDA	I2CSCL
Access	R	R	R	R	R	R	R	R	R	R	R	R	-	-	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register indicates the level of SDA and SCL of the I^2C bus.

[Description of Bits]

- **I2CSCL** (bit 0)
  Monitors the level of the SCL line.

- **I2CSDA** (bit 1)
  Monitors the level of the SDA line.

- **DBMON1** (bit7-4)
  Monitors the status of SCL bit counter.

- **DBMON2** (bit 11-8)
  Monitors the status of the SCL state machine.

- **DBMON3** (bit 15-12)
  Monitors the status of the main state machine.

[Note]
This bit is for debugging. Do not manipulate it for normal operation.

12.2.6 I^2C Bus Transfer Rate Setup Counter (I2CBC0, 1)

Address: 0x4008_3014 (0ch), 0x4008_3414 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	I2CBC[6:0]						
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register sets the count value for the counter that generates the transfer timing of the I^2C bus from the system clock. Normally, it is used to generate SCL/SDA.

[Description of Bits]

- **I2CBC** (bit 6-0)
  The relationship between the setting value of I2CBC and the transfer rate of the I^2C bus is as follows:
  I^2C bus transfer rate [bps] = System clock frequency x 1.05 / (I2CBC setting value x 8 + 6)
  Therefore,
  I2CBC = System clock frequency x 1.05 / (I^2C bus transfer rate [bps] x 8 + 6)

  The following table gives examples of setting values:

System clock frequency	I2CBC	
	100 kbps (I2CMD = “00”)	400 kbps (I2CMD = “01”)
32MHz	42 (0x2A)	10 (0x0A)

If “0” is set in the I2CBC register, the timing generation counter stops.

[Note]
- Set the system clock to 32 MHz.
- Set the I2CBC register above before setting the I2CCTL register.

12.2.7 I²C Mode Register (I2CMOD0, 1)

Address: 0x4008_3018 (0ch), 0x4008_3418 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	I2CBMEN
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register selects whether or not to use the buffer mode for the master transfer (master transmission and master reception).

[Description of Bits]

- **I2CBMEN** (bit 0)
  Set this bit to “1” to use I²C in the buffer mode.

I2CBMEN	Description
0	Buffer mode is not used
1	Buffer mode is used

[Note]
- Set this register in the initial setting flow or before the master transfer start. Operation cannot be guaranteed if the value of this register is changed during transfer.

12.2.8 I²C Buffer Mode Slave Address Register (I2CBUFSLV0, 1)

Address: 0x4008_301C (0ch), 0x4008_341C (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	I2CBMSLV[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
I2CBUFSLV register sets the slave address of the transfer destination device. This register is enabled only when the buffer mode is used.

[Description of Bits]

- **I2CBMSLV** (bit 15-0)
  Sets the slave address of the transfer destination device.
  Set I2CBMSLV7-1 to 7-bit address. Set I2CBMSLV15-8 and 0 to "0".

	15	14	13	12	11	10	9	8
I2CBUFSLV	“0”	“0”	“0”	“0”	“0”	“0”	“0”	“0”

	7	6	5	4	3	2	1	0
I2CBUFSLV	A7	A6	A5	A4	A3	A2	A1	“0”

[Note]
Set this bit in the initial setting flow or before the master transfer start.
Operation cannot be guaranteed if the value of this bit is changed during transfer.

12.2.9 I²C Buffer Mode Sub Address Register (I2CBUFSUB0, 1)

Address: 0x4008_3020 (0ch), 0x4008_3420 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	I2CBMSUB3[7:0]								I2CBMSUB2[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	I2CBMSUB1[7:0]								I2CBMSUB0[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
I2CBUFSUB register sets the sub address sent to the transfer destination device. This register is enabled only when the buffer mode is used.

[Description of Bits]

- **I2CBMSUB0** (bit 7-0)
  This register sets the sub address 0 sent to the transfer destination device.

- **I2CBMSUB1** (bit 15-8)
  This register sets the sub address 1 sent to the transfer destination device.

- **I2CBMSUB2** (bit 23-16)
  This register sets the sub address 2 sent to the transfer destination device.

- **I2CBMSUB3** (bit 31-24)
  This register sets the sub address 3 sent to the transfer destination device.

When the value set for I2CBMSL of the I2CBUFFOR register is “000”, the setting value of this register is disabled, and the sub address is not sent.
When the setting value of I2CBMSL is “001”, the data in I2CBMSUB0 is sent to the I²C bus.
When the setting value of I2CBMSL is “010”, the data in I2CBMSUB1 and I2CBMSUB0 is sent to the I²C bus in this order.
When the setting value of I2CBMSL is “011”, the data in I2CBMSUB2, I2CBMSUB1, and I2CBMSUB0 is sent to the I²C bus in this order.
When the setting value of I2CBMSL is “100”, the data in I2CBMSUB3, I2CBMSUB2, I2CBMSUB1, and I2CBMSUB0 is sent to the I²C bus in this order.

[Note]
Set this bit in the initial setting flow or before the master transfer start.
Operation cannot be guaranteed if the value of this bit is changed during transfer.

12.2.10 I²C Buffer Mode Format Register (I2CBUFFOR0, 1)

Address: 0x4008_3024 (0ch), 0x4008_3424 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	I2CBMDL[5:0]						I2CBMRW	I2CBMSL[2:0]		
Access	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
The I2CBUFFOR register sets the communication format in the buffer mode. Set the data length of the sent sub address, the data transfer direction, and the byte count of the transferred data. This register is enabled only when the buffer mode is used.

[Description of Bits]

- **I2CBMSL** (bit 2-0)
  Sets the data length of the transferred sub address in the buffer mode. “000” to “100” can be set.
  When it is set to “000”, the sub address is not sent.
  When it is set to a value between “001” and “100”, the sub address set for the I2CBUFSUB register is sent.
  When this bit is set to more than “101”, it will be set to “000”.

- **I2CBMRW** (bit 3)
  Sets the data transfer direction in the buffer mode.
  “1” and “0” indicate data reception and data transmission respectively.

- **I2CBMDL** (bit 9-4)
  This register sets the transferred byte count in the buffer mode. 0 to 32 bytes (“00000” to “100000”) can be set. When this bit is set to more than “100001”, it will be set to “000000”. When starting transfer with this bit set to “000000”, I2CBMDZ will be set to 1, and transfer will not be started.

[Note]
Set this bit in the initial setting flow or before the master transfer start.
Operation cannot be guaranteed if the value of this bit is changed during transfer.

12.2.11 I^2C Buffer Mode Control Register (I2CBUFCTL0, 1)

Address: 0x4008_3028 (0ch), 0x4008_3428 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	I2CBMSTA
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
The I2CBUFCTL register specifies the start of I2C operation in the buffer mode. This register is enabled only when the buffer mode is used.

[Description of Bits]

- **I2CBMSTA** (bit 0)
  Sets the start of the transfer in the buffer mode.
  For master transmission, set the I2CBUFSLV, I2CBUFSUB, I2CBUFFOR, I2CBUFMSK, and I2CBUFTMR registers, write the transmit data to the buffer, and then set this bit to “1”. This bit will be cleared to “0” after starting transfer when the transfer of the specified bytes is finished, or transfer is stopped due to errors such as NACK reception, unexpected STOP condition, and timeout.
  If the number of transferred bytes is different from the value of the buffer mode level register, the transmission does not start. This bit will be cleared to “0”, and an I2CBMAG interrupt will occur. If the number of transferred bytes is “0”, the transmission and reception does not start. This bit will be cleared to ”0”, and an I2CBMDZ interrupt will occur.

I2CBMSTA	Description
0	Transfer in the I2C buffer mode is stopped
1	Transfer in the I2C buffer mode is started

12.2.12 I^2C Buffer Mode Interrupt Mask Register (I2CBUFMSK0, 1)

Address: 0x4008_302C (0ch), 0x4008_342C (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	I2CBMDZIE	I2CBMAGIE	I2CBMISIE	I2CBMTOIE	I2CBMNAIE	-*	I2CBMFIIE
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	-	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
The I2CBUFMSK register controls each interrupt signal in the buffer mode. The corresponding bit of the status register is not controlled. (The status register will change even if an interrupt is disabled.)

[Description of Bits]

- **I2CBMFIIE** (bit 0)
  Specifies to enable or disable an I2CBMFI interrupt.
  The I2CBMFI status will change even if an interrupt is disabled by this bit.

I2CBMFIIE	Description
0	Disables I2CBMFI interrupt.
1	Enables I2CBMFI interrupt.

- **I2CBMNAIE** (bit 2)
  Specifies to enable or disable an I2CBMNA interrupt.
  The I2CBMNA status will change even if an interrupt is disabled by this bit.

I2CBMNAIE	Description
0	Disables I2CBMNA interrupt.
1	Enables I2CBMNA interrupt.

- **I2CBMTOIE** (bit 3)
  Specifies to enable or disable an I2CBTO interrupt.
  The I2CBMTO status will change even if an interrupt is disabled by this bit.

I2CBMTOIE	Description
0	Disables I2CBMTO interrupt.
1	Enables I2CBMTO interrupt.

- **I2CBMISIE** (bit 4)
  Specifies to enable or disable an I2CBMIS interrupt.
  The I2CBMIS status will change even if an interrupt is disabled by this bit.

I2CBMISIE	Description
0	Disables I2CBMIS interrupt.
1	Enables I2CBMIS interrupt.

- **I2CBMAGIE** (bit 5)
  Specifies to enable or disable an I2CBMAG interrupt.
  The I2CBMAG status will change even if an interrupt is disabled by this bit.

I2CBMAGIE	Description
0	Disables I2CBMAG interrupt.
1	Enables I2CBMAG interrupt.

- **I2CBMDZIE** (bit 6)
  Specifies to enable or disable an I2CBMDZ interrupt.
  The I2CBMDZ status will change even if an interrupt is disabled by this bit.

I2CBMDZIE	Description
0	Disables I2CBMDZ interrupt.
1	Enables I2CBMDZ interrupt.

[Note]
Set this bit in the initial setting flow or before the master transfer start.
Operation cannot be guaranteed if the value of this bit is changed during transfer.

12.2.13 I²C Buffer Mode Status Register (I2CBUFSTA0, 1)

Address: 0x4008_3030 (0ch), 0x4008_3430 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	I2CBMDZ	I2CBMAG	I2CBMIS	I2CBMTO	I2CBMNA	-*	I2CBMFI
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	-	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. “0” is read when reading. Write “0” when writing.

[Description of Register]
The I2CBUFSTA register indicates each status in the buffer mode. Write 0 to each bit to clear it. This register is enabled only when the buffer mode is used.

[Description of Bits]

- **I2CBMFI** (bit 0)
  It is set to “1” when this I2C module finishes the transfer in the buffer mode.
  This bit is cleared by writing “0” by software.

I2CBMFI	Description
0	Transfer in the buffer mode is not completed
1	Transfer in the buffer mode is completed

- **I2CBMNA** (bit 2)
  It is set to “1” when it receives a NACK and the transfer is finished.
  This bit is cleared by writing “0” by software.

I2CBMNA	Description
0	Has not received a NACK.
1	Received a NACK and terminated transfer.

- **I2CBMTO** (bit 3)
  It is set to “1” when a timeout occurs before the end of the transfer, and the transfer ends abnormally. For example, when SCL prolonging by the slave device is not terminated in a certain time, this bit is set to “1”.
  This bit is cleared by writing “0” by software.

I2CBMTO	Description
0	Timeout has not occurred
1	Timeout has occurred

- **I2CBMIS** (bit 4)
  This bit is set to "1" if a STOP condition occurs at unexpected timing, and the transfer ends abnormally. This bit is set to "1" if a STOP condition is detected before this I2C device transmits a STOP condition during a transfer in the buffer mode.
  This bit is cleared by writing "0" by software.

I2CBMIS	Description
0	Has not detected an unexpected STOP condition
1	Detected an unexpected STOP condition.

- **I2CBMAG** (bit 5)
  This bit is set to "1" if the number of transferred bytes set in I2CBMDL and the value of I2CBML do not match at the start of the transmission (when I2CBMRW is set to "0" and "1" is written to I2CBMSTA). In this case, the transfer is not started.
  This bit is cleared by writing "0" by software.

I2CBMAGIE	Description
0	The number of transferred bytes and the buffer capacity match at the start of the transmission
1	The transmission has not been started because the number of transferred bytes and the buffer capacity do not match at the start of the transmission

- **I2CBMDZIE** (bit 6)
  This bit is set to "1" if the number of transferred bytes set in I2CBMDL is 0 at the start of the transmission (when "1" is written to I2CBMSTA). In this case, the transfer is not started.
  This bit is cleared by writing "0" by software.

I2CBMDZ	Description
0	The number of transferred bytes is not 0 at the start of transmission
1	The transmission has not been started because the number of transferred bytes is 0 at the start of transmission

12.2.14 I²C Buffer Mode Level Register (I2CBUFLEV0, 1)

Address: 0x4008_3034 (0ch), 0x4008_3434 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*			I2CBML[5:0]			
Access	-	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. “0” is read when reading. Write “0” when writing.

[Description of Register]
The I2CBUFLEV register indicates the amount of data remaining in the buffer. This register is enabled only when the buffer mode is used.

12.2.15 I²C Timer Register (I2CTMR0, 1)

Address: 0x4008_3048 (0ch), 0x4008_3448 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	I2CT[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
The I2CTMR register sets the interval of the timeout in the buffer mode. This register is enabled only when the buffer mode is used.
The I2CBMTO bit in the I2CBUFSTA register is set to “1” when it takes the timeout setting value or longer to transmit and receive each one byte in the buffer mode. The transfer is stopped when a timeout occurs.

[Description of Bits]

- **I2CT[15:0]** (bits 15-0)
  The timeout interval can be calculated from the setting value of I2CT as follows:

  Timeout interval = (I2CT setting value * 8) / System clock frequency

  Here are the examples of setting the timeout interval to 1ms and 8ms.

System clock frequency	I2CT	
	For 1ms interval	For 8ms interval
32MHz	0x0FA0	0x7D00

  When I2CT is set to ”0”, a timeout interrupt does not occur.

[Note]
Set this bit in the initial setting flow or before the master transfer start.
Operation cannot be guaranteed if the value of this bit is changed during transfer.
Setting I2CT to less than the time required for the transfer always results in a timeout.
One-byte transfer takes 90 µs in the standard mode (100 kbps) and 22.5 µs in the fast mode (400 kbps).

12.2.16 I²C Input Noise Filter Setting Register (I2CNF0, 1)

Address: 0x4008_3050 (0ch), 0x4008_3450 (1ch)
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	NSFLCLK	-*	-*	-*	I2CNSFLON
Access	-	-	-	-	-	-	-	-	-	-	-	R/W	-	-	-	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
The I2CNF register sets whether or not to use the noise filter for SCL and SDA input.

[Description of Bits]

- **I2CNSFLON** (bit 0)
  Sets whether or not to use the input noise filter for I2C.
  The initial value is "1", in which case the noise filter is used.

I2CNSFLON	Description
0	Noise filter is not used
1	Noise filter is used

- **NSFLCLK** (bit 4)
  Selects a clock used for the input noise filter.
  1/1 x system clock (when system clock = 32 MHz) functions as a 62.5 ns filter.

I2CBMNA	Description
0	1/1 x system clock is used for the noise filter
1	1/2 x system clock is used for the noise filter

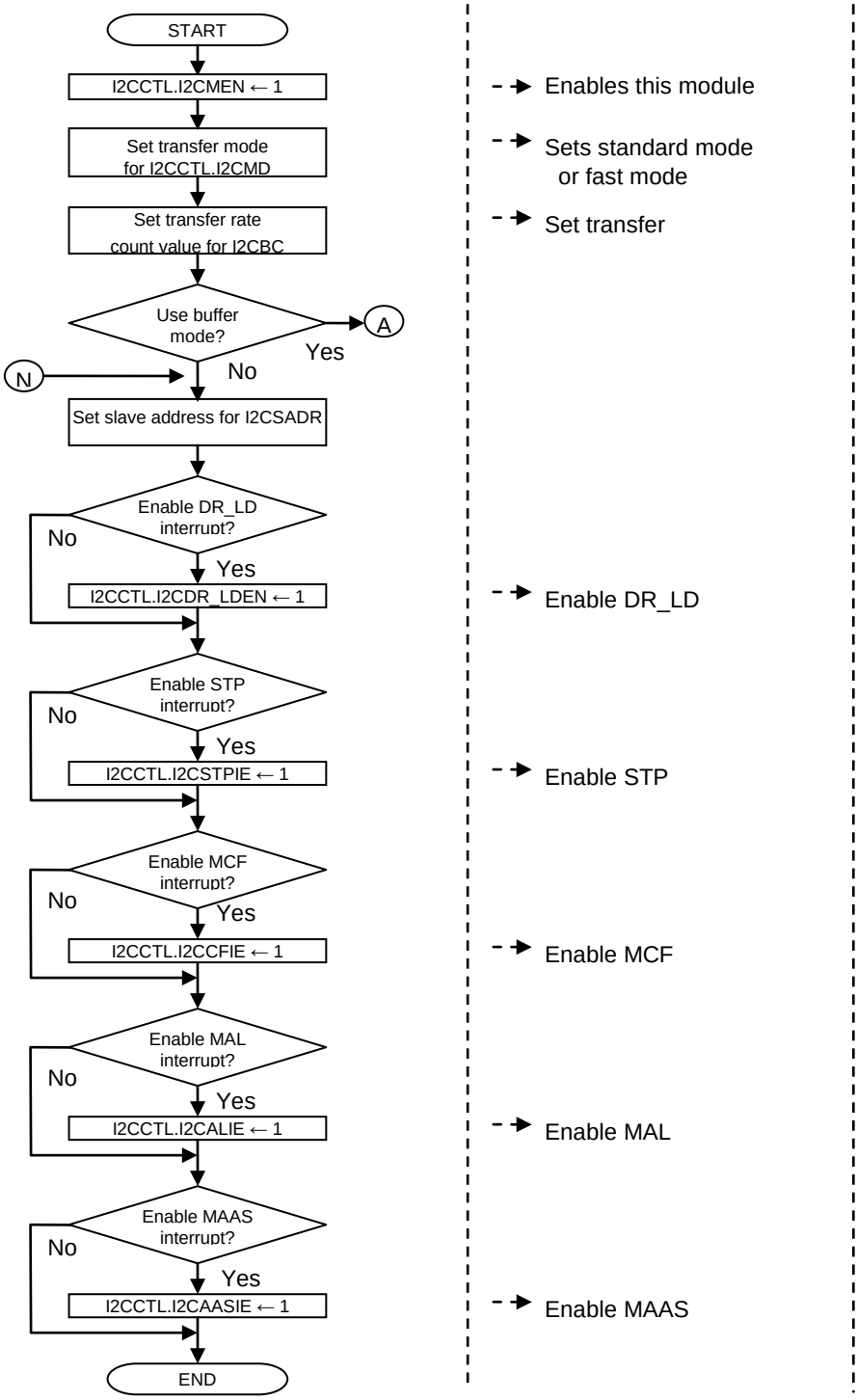
Table 12-5 Examples of NSFLCLK Setting Values

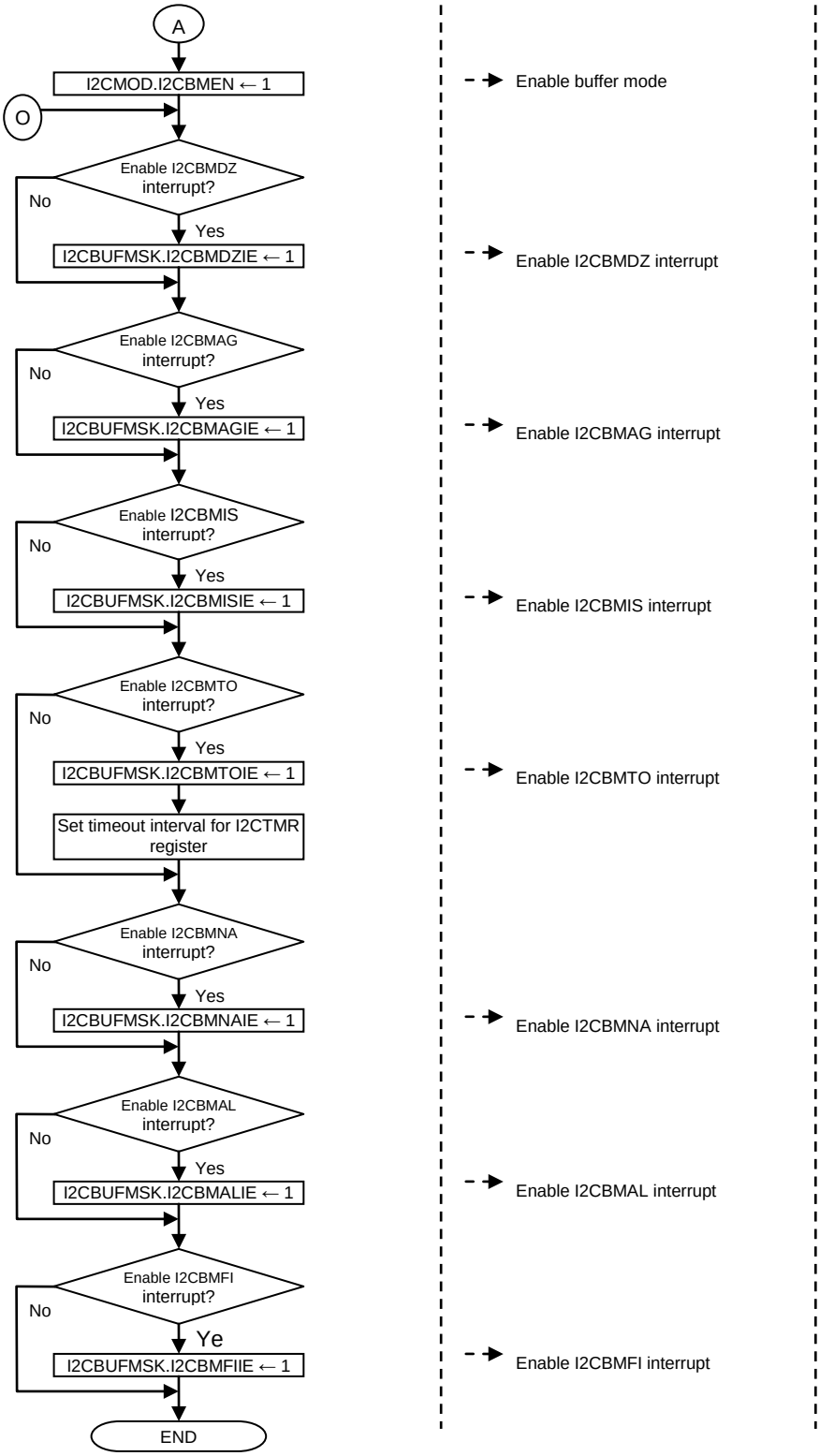
System clock frequency	When I2CNSFLCLK = 0	When I2CNSFLCLK = 1
32MHz	62.5ns	125ns

12.3 Description of Operation

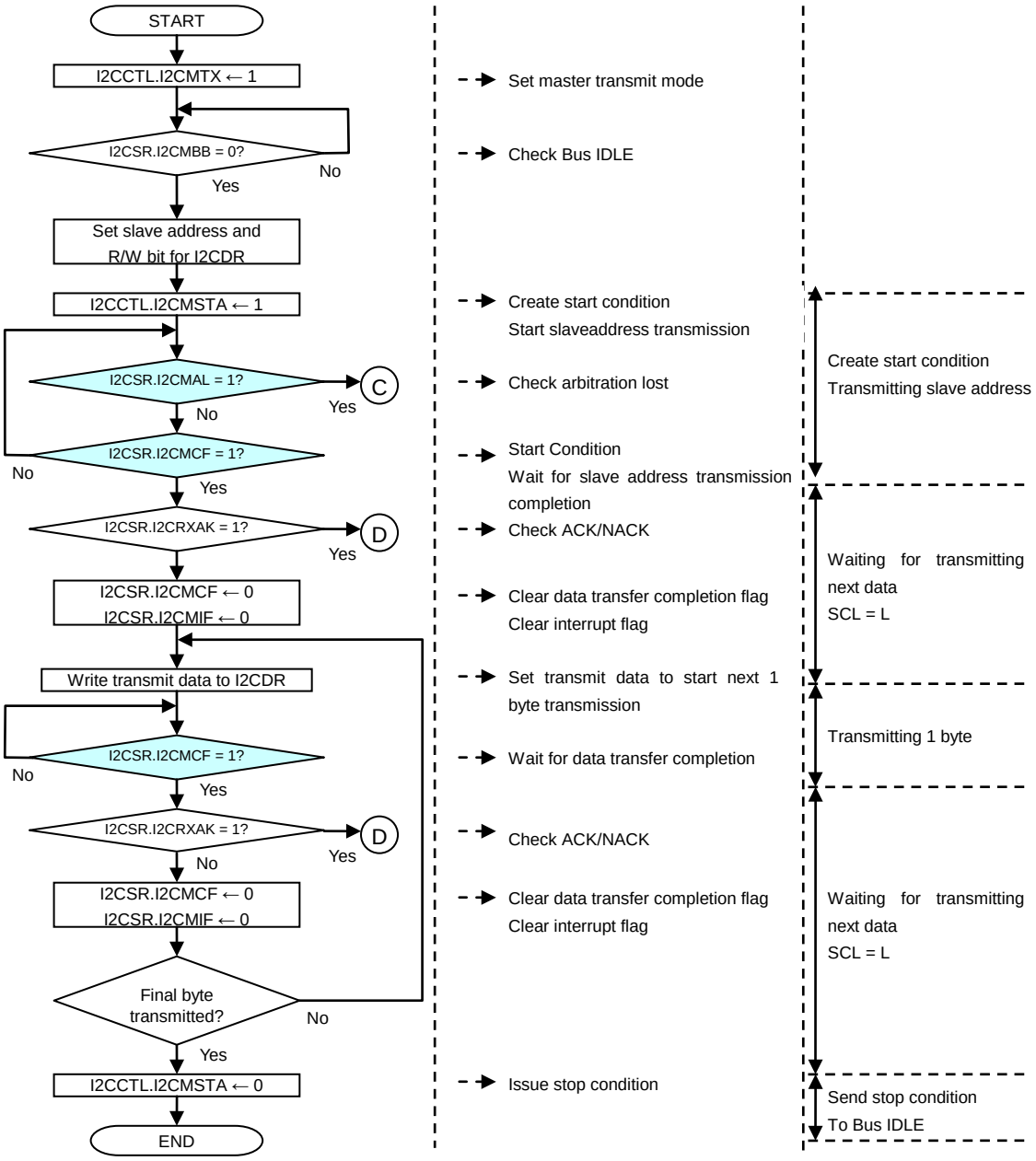
12.3.1 Flow of Initial Setting

This section shows the initial setting flow. "bit name ← 1(0)" represents that software writes 1(0) to this bit. The shaded portions show interrupt sources.

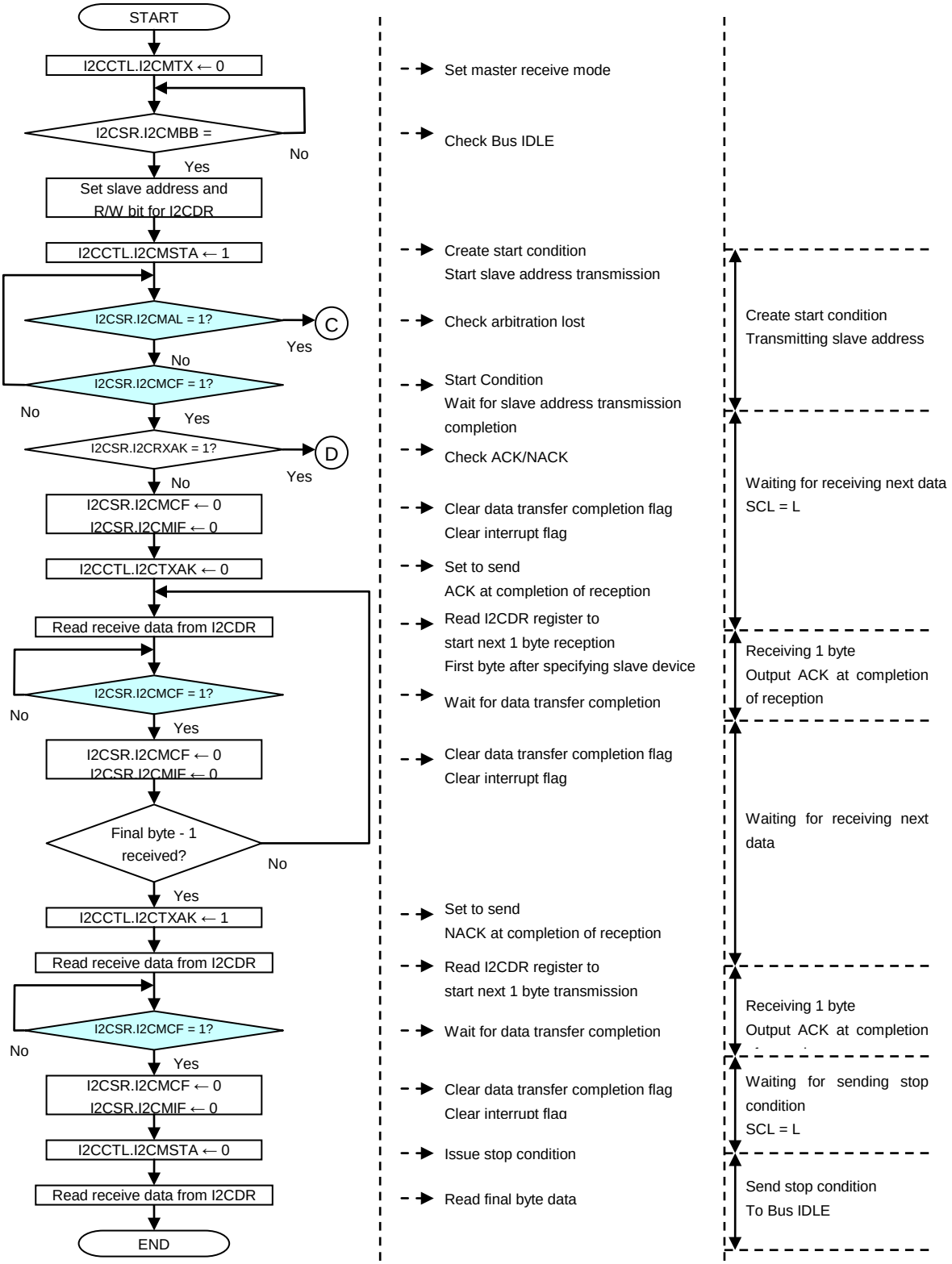



A
I2CMOD.I2CBMEN ← 1
O
Enable I2CBMDZ interrupt?
No
Yes
I2CBUFMSK.I2CBMDZIE ← 1
Enable I2CBMAG interrupt?
No
Yes
I2CBUFMSK.I2CBMAGIE ← 1
Enable I2CBMIS interrupt?
No
Yes
I2CBUFMSK.I2CBMISIE ← 1
Enable I2CBMTO interrupt?
No
Yes
I2CBUFMSK.I2CBMTOIE ← 1
Set timeout interval for I2CTMR register
Enable I2CBMNA interrupt?
No
Yes
I2CBUFMSK.I2CBMNAIE ← 1
Enable I2CBMAL interrupt?
No
Yes
I2CBUFMSK.I2CBMALIE ← 1
Enable I2CBMFI interrupt?
No
Ye
I2CBUFMSK.I2CBMFIIE ← 1
END
Enable buffer mode
Enable I2CBMDZ interrupt
Enable I2CBMAG interrupt
Enable I2CBMIS interrupt
Enable I2CBMTO interrupt
Enable I2CBMNA interrupt
Enable I2CBMAL interrupt
Enable I2CBMFI interrupt

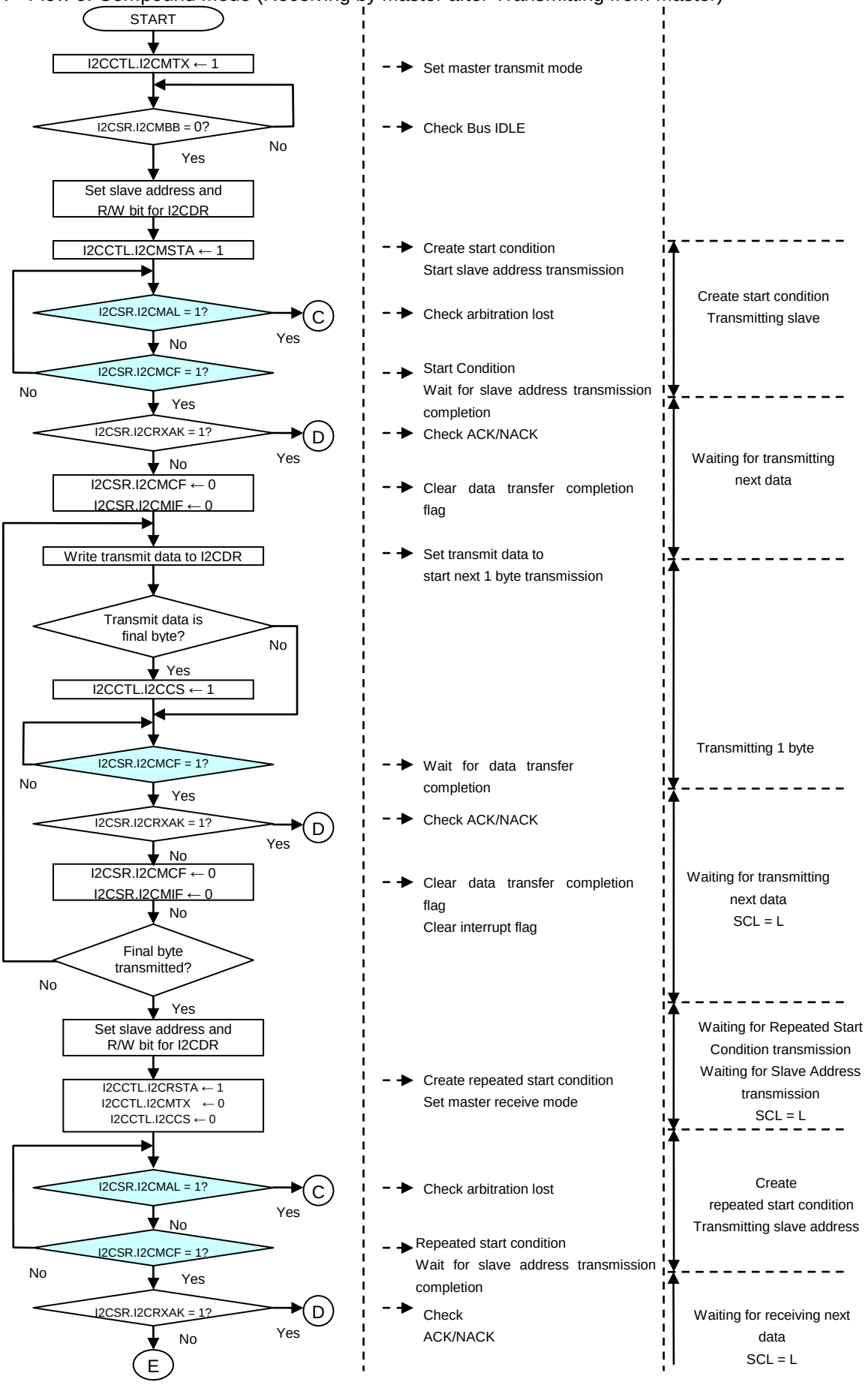
12.3.2 Flow of Master Transmission

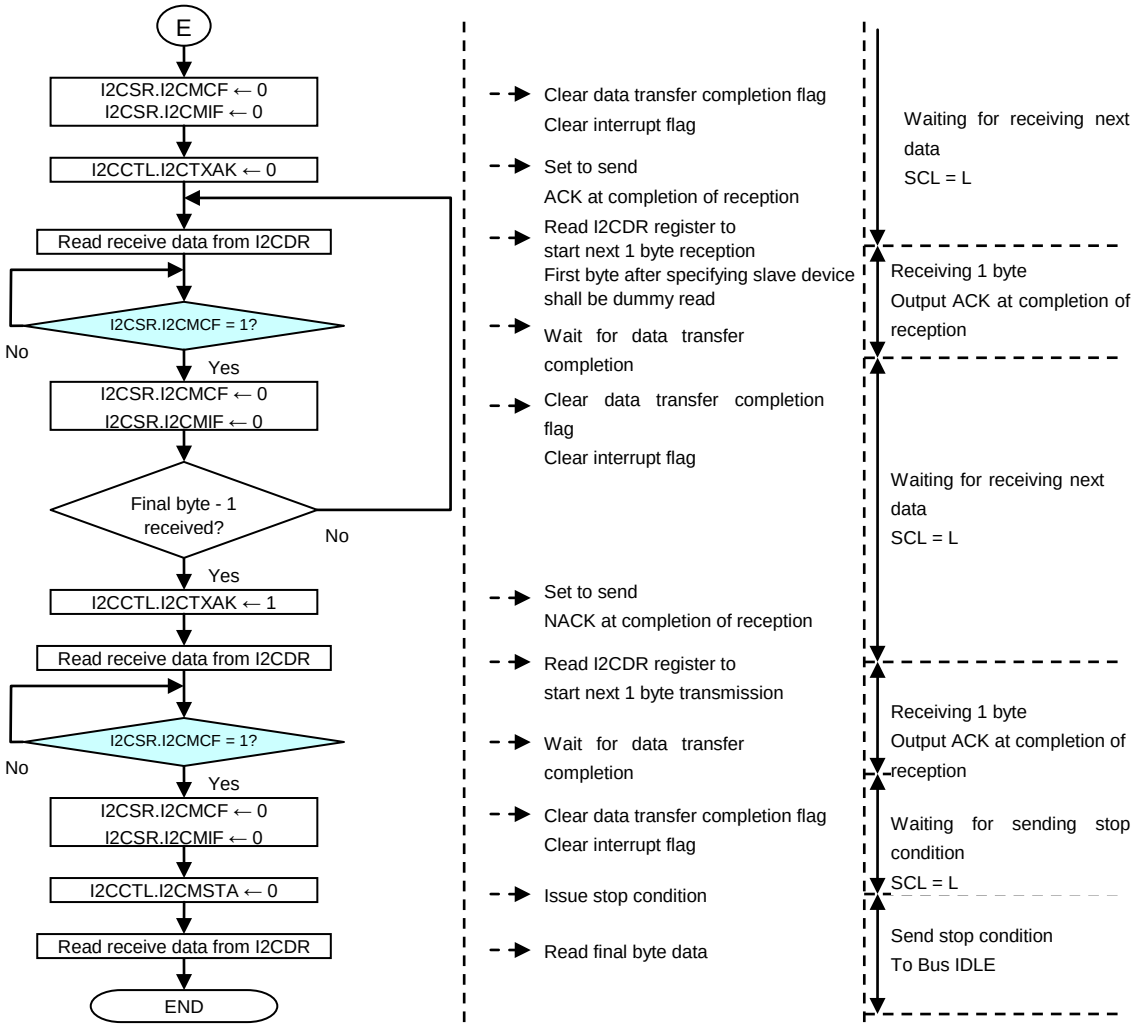


12.3.3 Flow of Master Reception

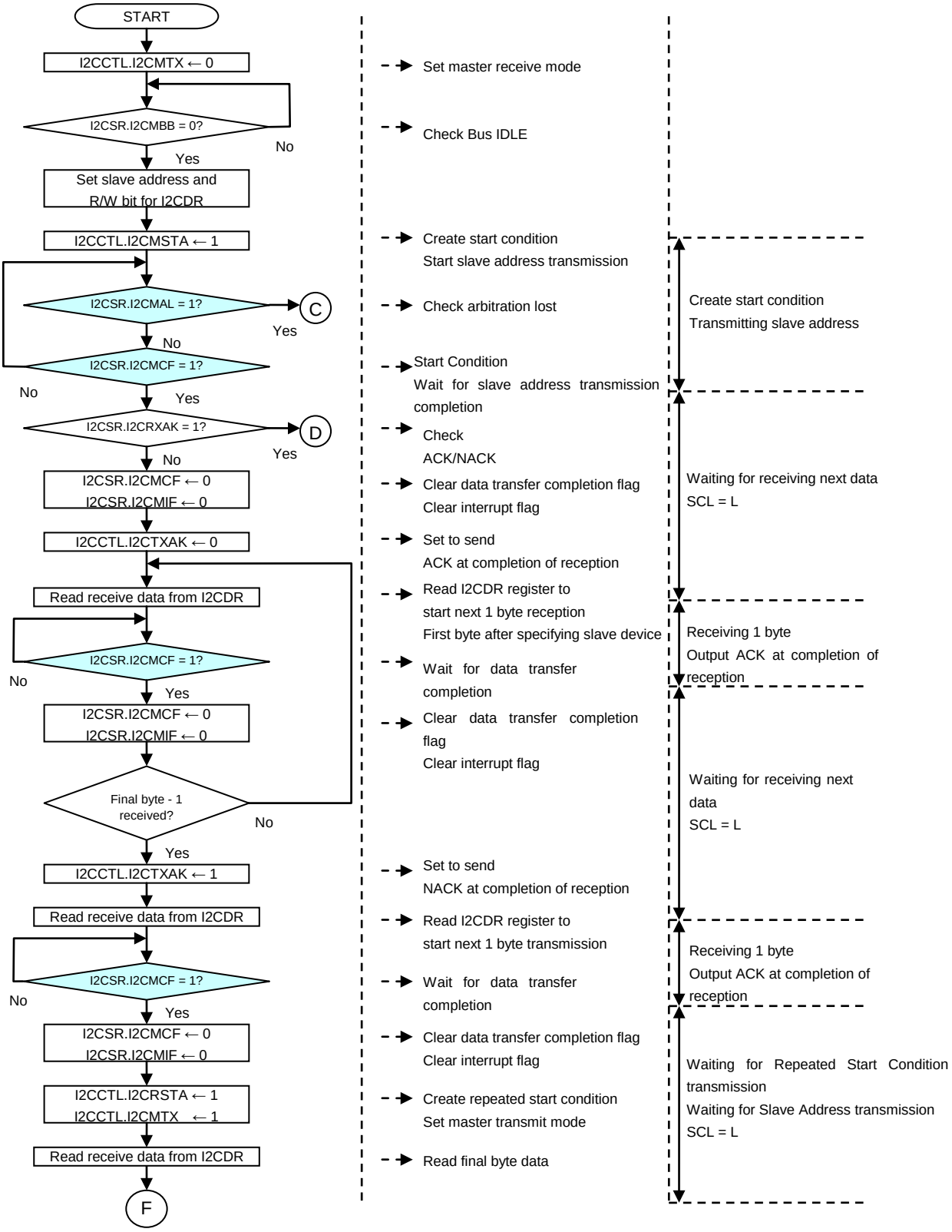


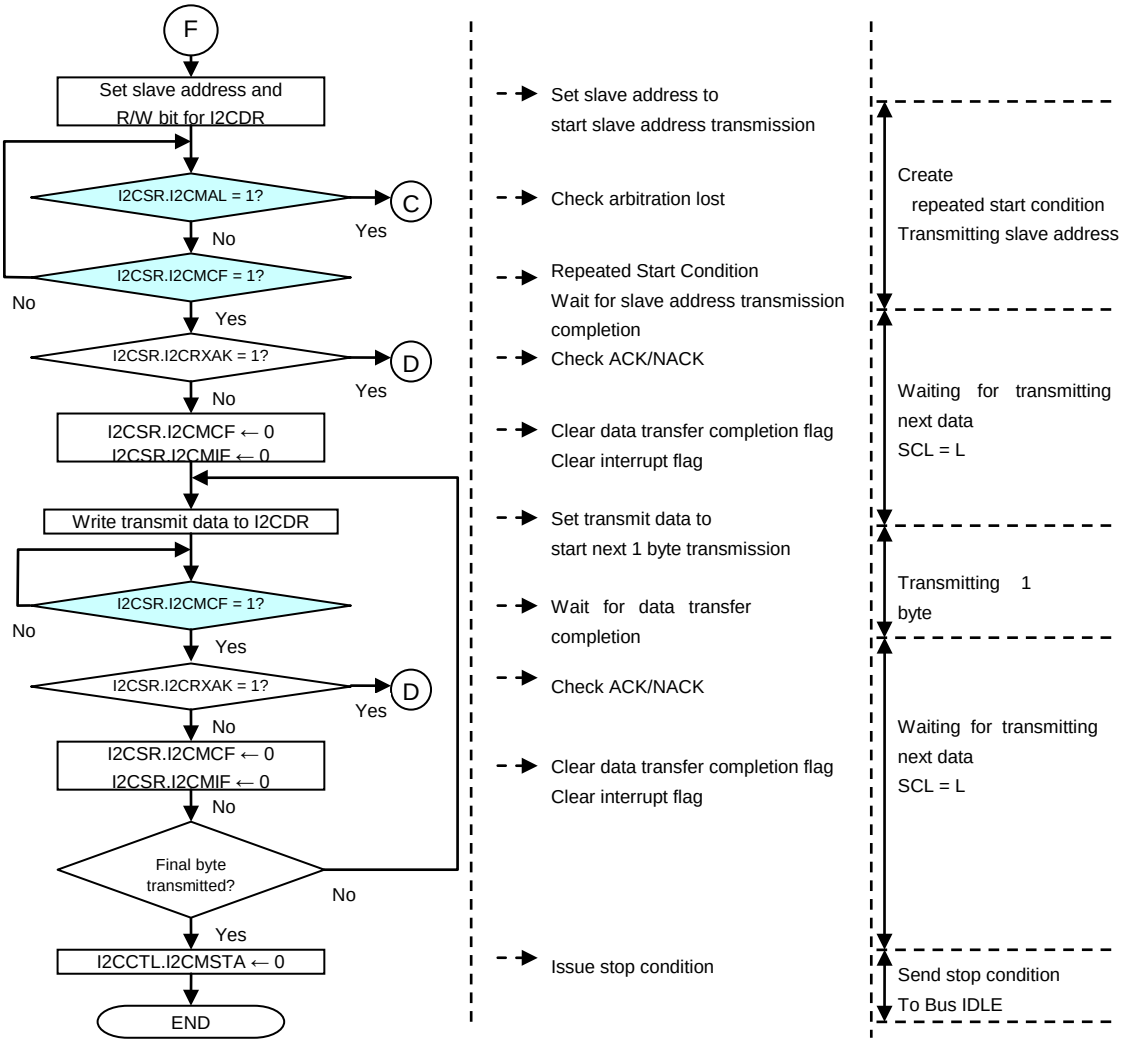
12.3.4 Flow of Compound Mode (Receiving by Master after Transmitting from Master)



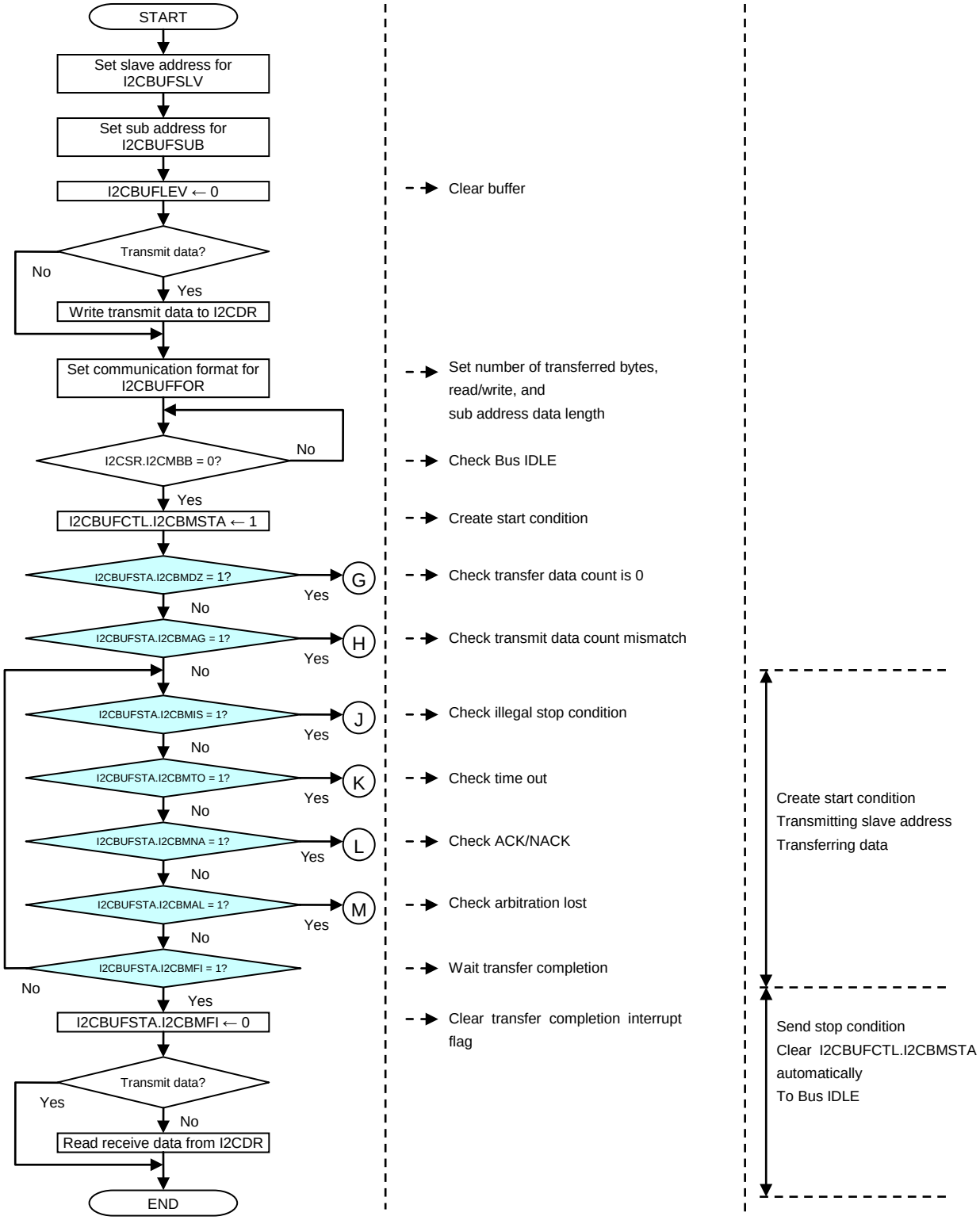

E
I2CSR.I2CMCF ← 0
I2CSR.I2CMIF ← 0
I2CCTL.I2CTXAK ← 0
Read receive data from I2CDR
I2CSR.I2CMCF = 1?
No
Yes
I2CSR.I2CMCF ← 0
I2CSR.I2CMIF ← 0
Final byte - 1 received?
No
Yes
I2CCTL.I2CTXAK ← 1
Read receive data from I2CDR
I2CSR.I2CMCF = 1?
No
Yes
I2CSR.I2CMCF ← 0
I2CSR.I2CMIF ← 0
I2CCTL.I2CMSTA ← 0
Read receive data from I2CDR
END
Clear data transfer completion flag
Clear interrupt flag
Set to send
ACK at completion of reception
Read I2CDR register to
start next 1 byte reception
First byte after specifying slave device
shall be dummy read
Wait for data transfer
completion
Clear data transfer completion
flag
Clear interrupt flag
Set to send
NACK at completion of reception
Read I2CDR register to
start next 1 byte transmission
Wait for data transfer
completion
Clear data transfer completion flag
Clear interrupt flag
Issue stop condition
Read final byte data
Waiting for receiving next
data
SCL = L
Receiving 1 byte
Output ACK at completion of
reception
Waiting for receiving next
data
SCL = L
Receiving 1 byte
Output ACK at completion of
reception
Waiting for sending stop
condition
SCL = L
Send stop condition
To Bus IDLE

12.3.5 Flow of Compound Mode (Transmitting from Master after Receiving by Master)




F
Set slave address and R/W bit for I2CDR
I2CSR.I2CMAL = 1?
C
Yes
No
I2CSR.I2CMCF = 1?
No
Yes
I2CSR.I2CRXAK = 1?
D
Yes
No
I2CSR.I2CMCF ← 0
I2CSR.I2CMIF ← 0
Write transmit data to I2CDR
I2CSR.I2CMCF = 1?
No
Yes
I2CSR.I2CRXAK = 1?
D
Yes
No
I2CSR.I2CMCF ← 0
I2CSR.I2CMIF ← 0
No
Final byte transmitted?
No
Yes
I2CCTL.I2CMSTA ← 0
END
Set slave address to start slave address transmission
Check arbitration lost
Repeated Start Condition
Wait for slave address transmission completion
Check ACK/NACK
Clear data transfer completion flag
Clear interrupt flag
Set transmit data to start next 1 byte transmission
Wait for data transfer completion
Check ACK/NACK
Clear data transfer completion flag
Clear interrupt flag
Issue stop condition
Create repeated start condition
Transmitting slave address
Waiting for transmitting next data
SCL = L
Transmitting 1 byte
Waiting for transmitting next data
SCL = L
Send stop condition
To Bus IDLE

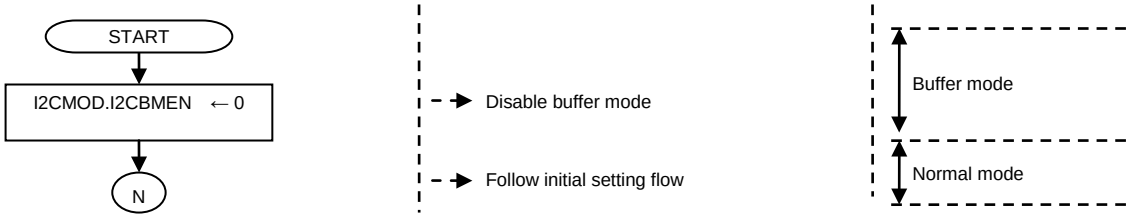
12.3.6 Flow When Using Buffer Mode



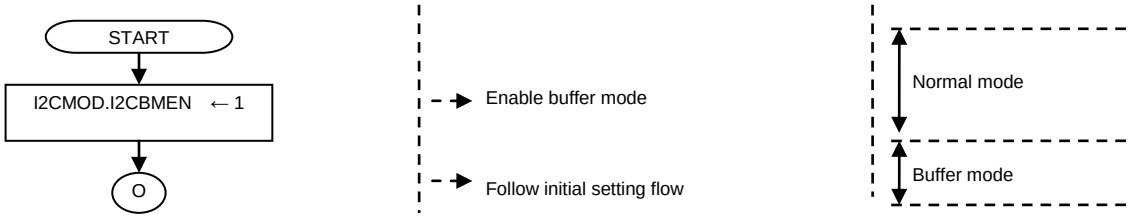
G
I2CBUFSTA.I2CBMDZ ← 0
Error END
Clear interrupt flag of transfer data count 0
Buffer mode error end
Clear
I2CBUFCTL.I2CBMSTA
automatically
H
I2CBUFSTA.I2CBMAG ← 0
Error END
Clear interrupt flag of transfer data mismatch
Buffer mode error end
Clear
I2CBUFCTL.I2CBMSTA
automatically
J
I2CBUFSTA.I2CBMIS ← 0
Error END
Clear illegal stop condition
interrupt flag
Buffer mode error end
Abnormal Bus end
Clear
I2CBUFCTL.I2CBMSTA
automatically
K
I2CBUFSTA.I2CBMTO ← 0
Error END
Clear time out
interrupt flag
Buffer mode error end
Abnormal Bus end
Clear
I2CBUFCTL.I2CBMSTA
automatically
L
I2CBUFSTA.I2CBMNA ← 0
Error END
Clear NACK
interrupt flag
Buffer mode error end
Send stop condition
To Bus IDLE
Clear
I2CBUFCTL.I2CBMSTA
automatically
M
I2CBUFSTA.I2CBMAL ← 0
Error END
Clear arbitration lost
interrupt flag
Buffer mode error end
Other master device
communicating
Clear
I2CBUFCTL.I2CBMSTA
automatically

12.3.7 Flow of Mode Switching

12.3.7.1 Flow of Switching to Normal Mode



12.3.7.2 Flow of Switching to Buffer Mode

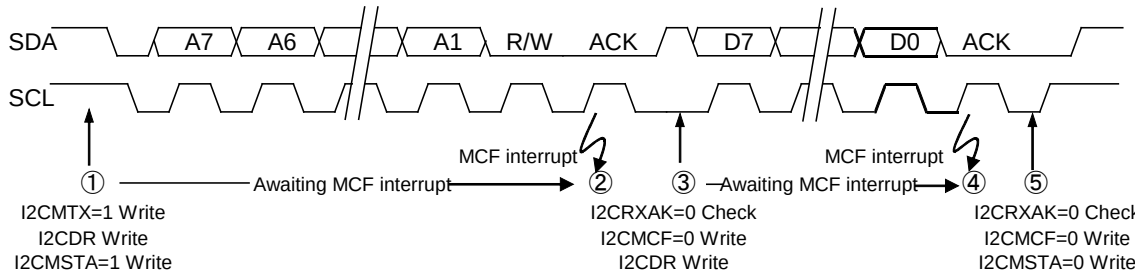


[Note] Be sure to switch modes before starting I2C transfer.

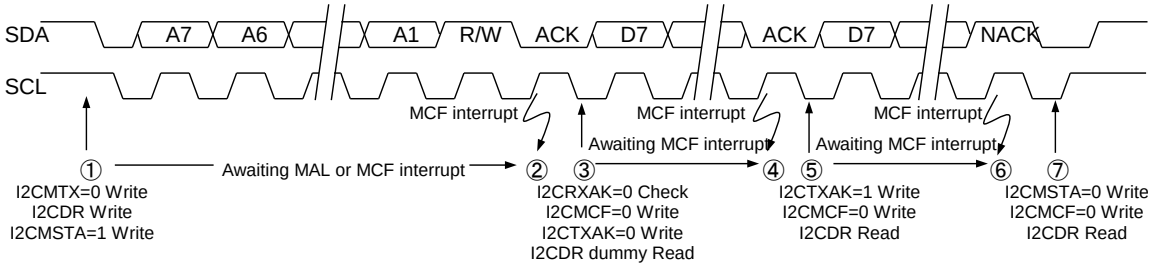
12.4 Waveform in Each Mode

In the figures below, the hatched portions are the segments that are driven by the transfer destination.

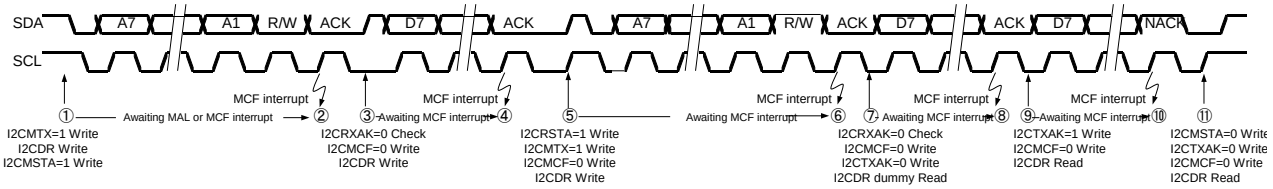
12.4.1 Waveform Transmitted by Master



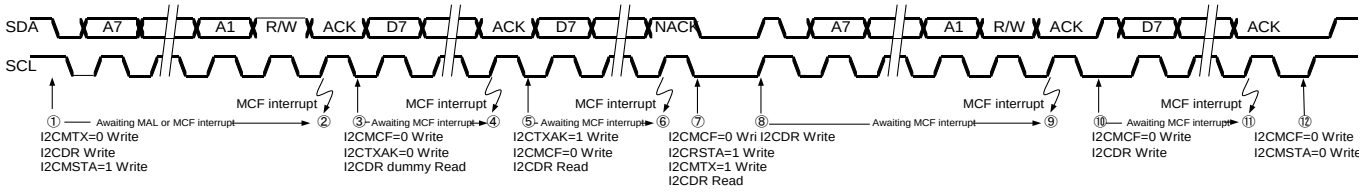
12.4.2 Waveform Received by Master



12.4.3 Waveform of Compound Format (Master Transmission + Master Reception)

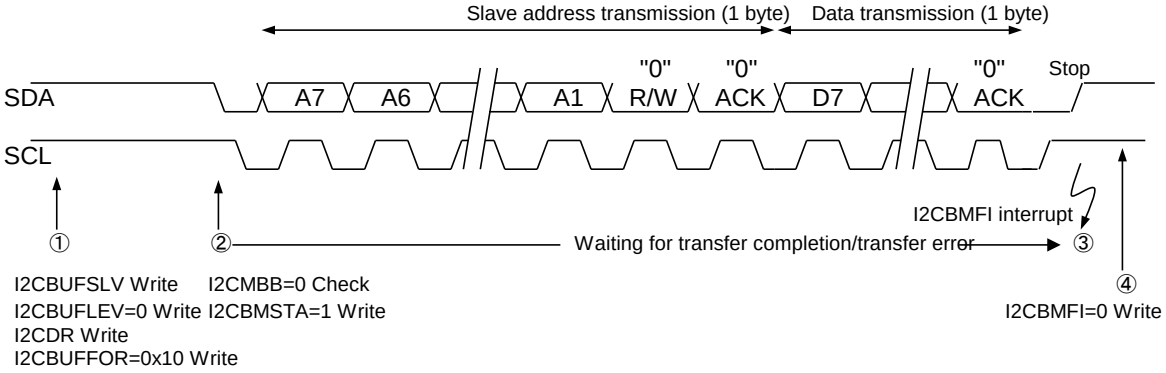


12.4.4 Waveform of Compound Format (Master Reception + Master Transmission)



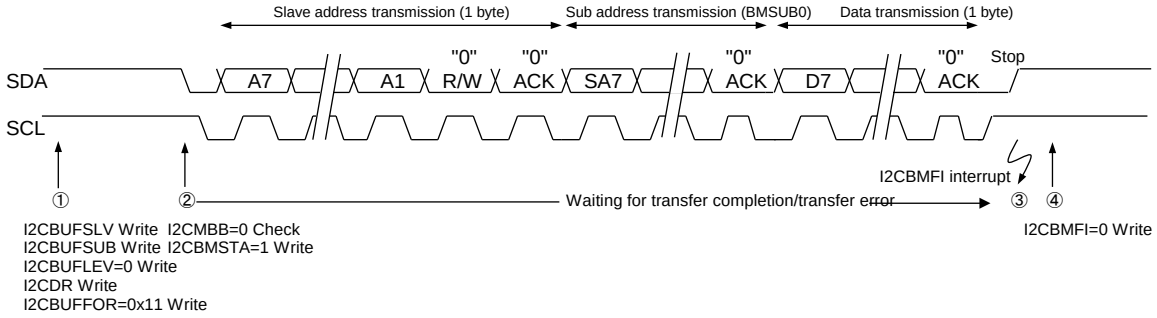
12.4.5 Waveform 1 When Using Buffer Mode

(When data length of sub address = 0, data transmission, number of transferred bytes = 1)



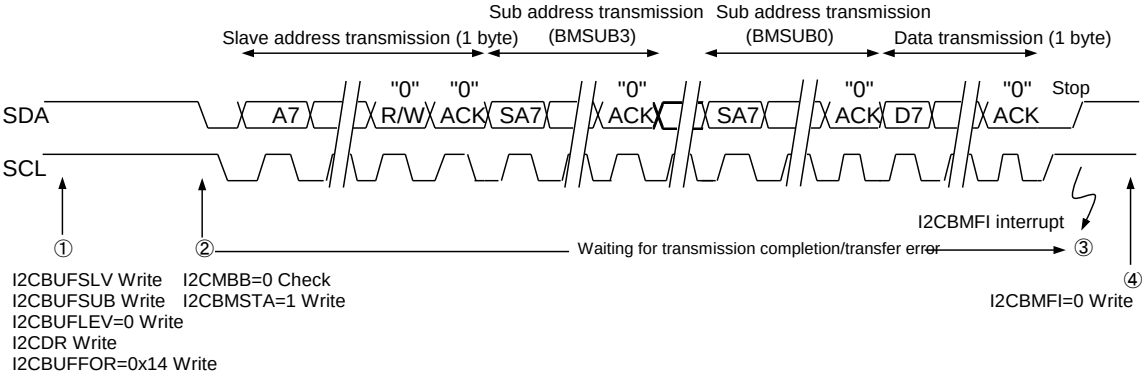
12.4.6 Waveform 2 When Using Buffer Mode

(When data length of sub address = 1, data transmission, number of transferred bytes = 1)



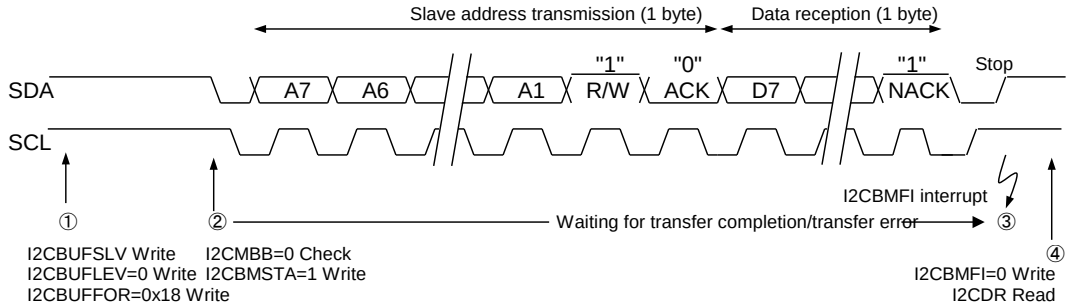
12.4.7 Waveform 3 When Using Buffer Mode

(When data length of sub address = 4, data transmission, number of transferred bytes = 1)



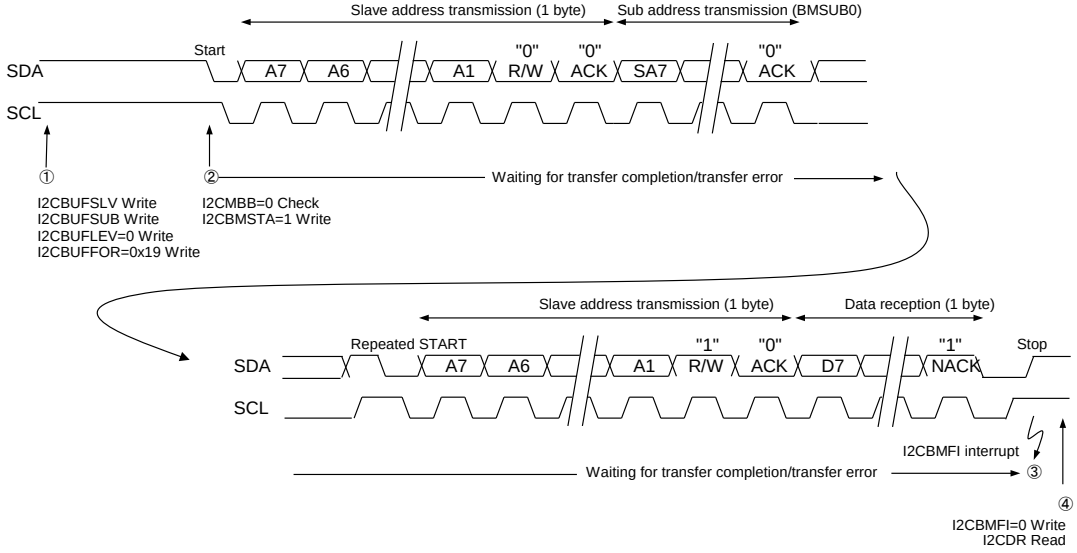
12.4.8 Waveform 4 When Using Buffer Mode

(When data length of sub address = 0, data reception, number of transferred bytes = 1)



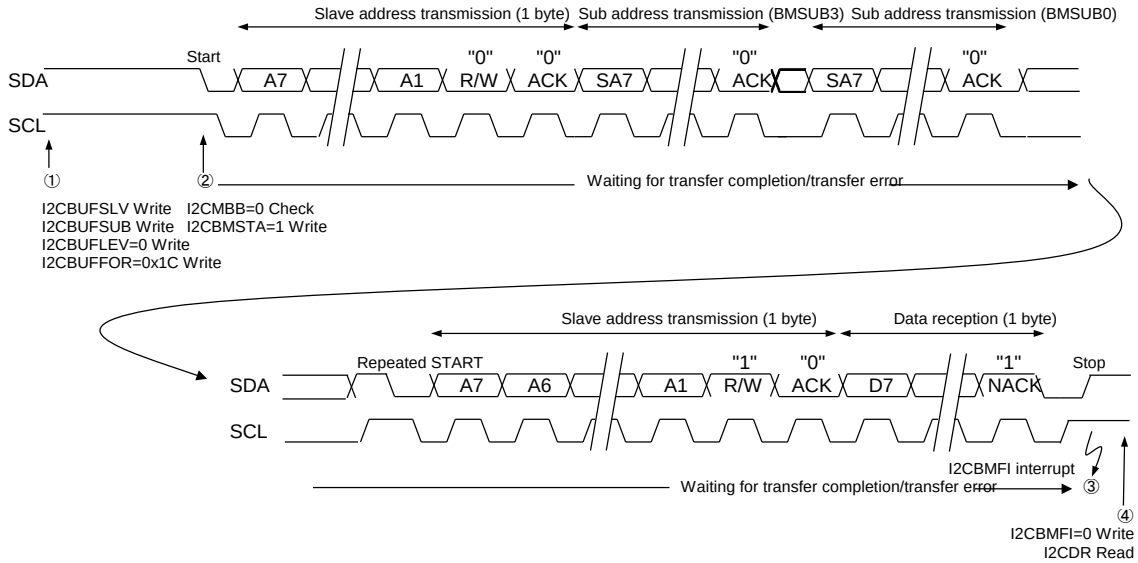
12.4.9 Waveform 5 When Using Buffer Mode

(When data length of sub address = 1, data reception, number of transferred bytes = 1)



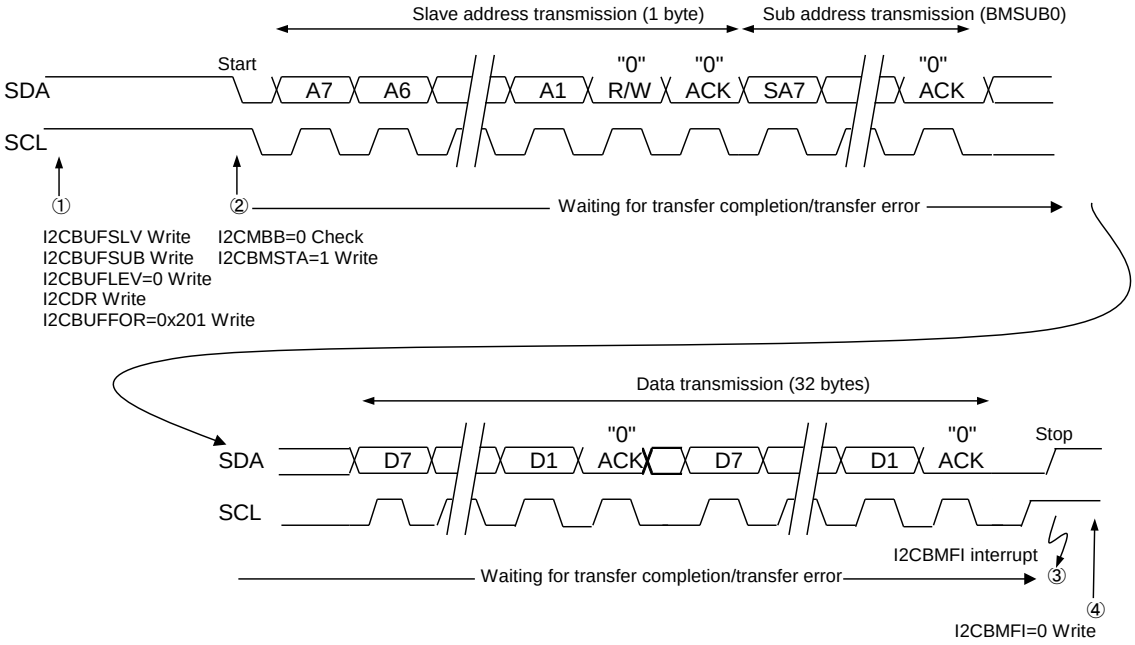
12.4.10 Waveform 6 When Using Buffer Mode

(When data length of sub address = 2, data reception, number of transferred bytes = 1)



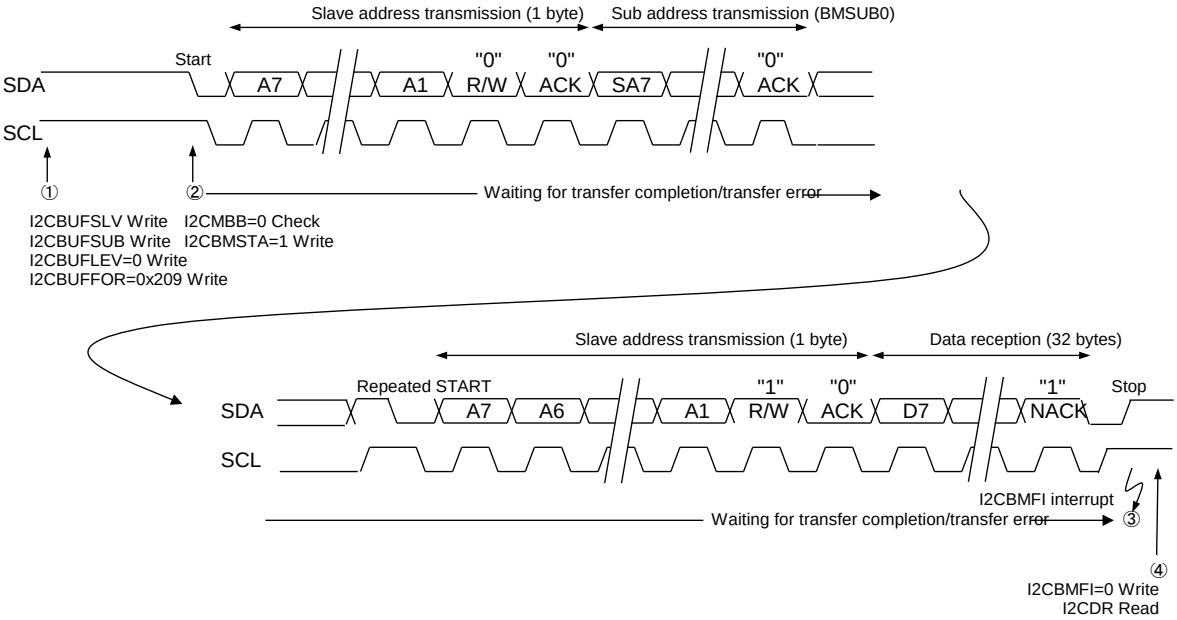
12.4.11 Waveform 7 When Using Buffer Mode

(When data length of sub address = 1, data transmission, number of transferred bytes = 32)



12.4.12 Waveform 8 When Using Buffer Mode

(When data length of sub address = 1, data reception, number of transferred bytes = 32)



12.5 Restrictions

[I2C Master transmission mode]

- If there is a NACK response after data is transferred in the I2C master transmit mode, a STOP condition is automatically sent at the same time it enters the IDLE state. To resume the transmission, it is necessary to set the MSTA of the control register to 1 and send a START condition again.

- In the I2C master transmit mode, be sure to write the first byte of the transmit data before setting MSTA=1 (sending START condition). (The data transmission is automatically started after sending START condition) Write subsequent bytes of the transmit data after the DR_LD status is set to 1. Any transmit data written to the data register with the DR_LD status set to 0 cannot be guaranteed.

- In the I2C master transmit mode, be sure to set the STOP condition (MSTA=0) after transferring the data (after the MCF status is set to 1). The STOP condition sending is started after the ACK response cycle. If it is set at any other timing than an ACK cycle, the STOP condition will be sent immediately, which may cause a failure.

12.6 Specifying Port Registers

To use I2C1, the applicable bit of each related port register needs to be set. See Chapter 15, "GPIO" for details about the port registers.

12.6.1 Operating I2C1

Set PA0 (bit 1, 0) of the PAMOD register to “01” to select the secondary function (SDA1_M) of PA0.
Set PA1 (bit 5, 4) of the PAMOD register to ”01”, and then select the secondary function of PA1 (SCL1_M).

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	*	*	*	*	*	*	*	*	**0**	**1**	*	*	**0**	**1**

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Set the bits 0 and 1 of the PIODIR register to “0” to set the input/output mode of PA0 to output.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	PIODIR						
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	**0**	**0**

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the I2C1 function

[Note]
The PIOCON register does not need to be set. When the secondary function is selected, the setting is automatically changed, but the value of the PIOCON register does not change.

Chapter 13 UART

13. UART

13.1 Overview

This LSI includes a UART with 16-byte transmit and receive FIFOs. This UART functions as the input/output interface, carries out serial-to-parallel conversion of the data sent from the peripheral devices, and also converts the parallel data sent from the CPU into serial data. In the FIFO mode, it is possible to store 16 bytes of data during transmission and reception. Further, the receive FIFO generates 3 bits of error data for every byte of received data. The CPU can read out the status of ACE at any time. The information that can be read out consists of the type and status of the transfer operation under execution, and the statuses of errors such as parity, overrun, or framing errors, or break interrupt, etc.

The use of UART requires setting of the secondary functions of Port A. For the secondary functions of Port A, see Chapter 5, "MCU Control Function".

13.1.1 Features

- Full duplex buffer system
- All status reporting function
- 16-byte transmit and receive FIFOs
- Independent control of transmit, receive, line status data set interrupt and FIFO
- Programmable serial interface
  - 5, 6, 7, or 8 bits per character
  - Odd parity, even parity, no parity generation and verification
  - 1, 1.5, or 2 stop bits
- Communication speed: Settings available in the range of 300 bps to 115200 bps

13.1.2 Configuration

Figure 13-1 shows the configuration diagram of the UART.

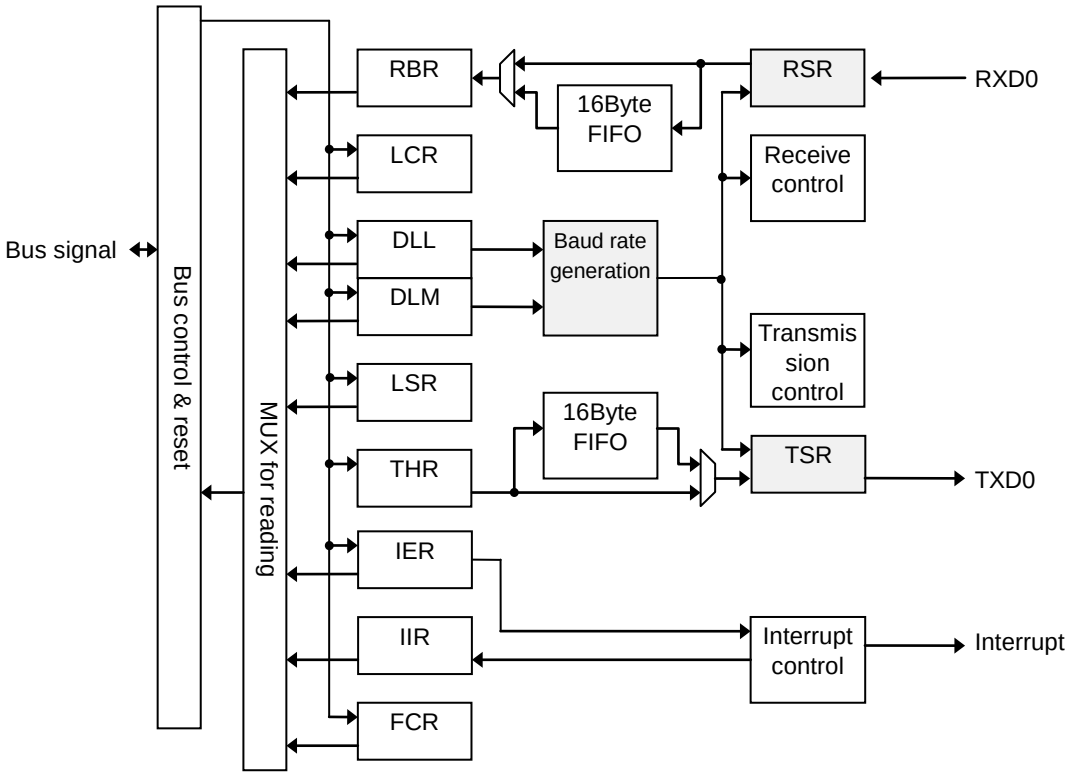


Figure 13-1 Configuration Diagram

13.1.3 List of Pins

Table 13-1 List of Pins Interfaced with the Outside of LSI

Pin name	I/O	Description
PA2/RXD0	I	UART data input pin Used for the secondary function.
PA3/TXD0	O	UART data output pin Used for the secondary function.

13.2 Description of Registers

13.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4008_1000	UART receive data register UART transmit data register UART baud rate dividing register (LSB)	UARTRBR UARTTHR UARTDLL	R W R/W	32	Undefined Undefined 0x0000_0000
0x4008_1004	UART interrupt enable register UART baud rate dividing register (MSB)	UARTIER UARTDLM	R/W R/W	32	0x0000_0000 0x0000_0000
0x4008_1008	UART interrupt status register UART FIFO control register	UARTIIR UARTFCR	R W	32	0x0000_0001 0x0000_0000
0x4008_100C	UART line control register	UARTLCR	R/W	32	0x0000_0000
0x4008_1014	UART line status register	UARTLSR	R	32	0x0000_0060
0x4008_101C	UART scratchpad register	UARTSCR	R/W	32	0x0000_0000

Note) R/W access to addresses 0x4008_1010 and 0x4008_1018 is prohibited. Proper operation cannot be guaranteed if accessed.

13.2.2 UART Receive Data Register / UART Transmit Data Register / UART Baud Rate Dividing Register(LSB) (UARTRBR, UARTTHR, UARTDLL)

Address: 0x4008_1000
Access: R/W
Access size: 32 Bits
Initial value: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	RBR[7:0]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	X	X	X	X	X	X	X	X

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
UARTRBR is a special function register (SFR) that provides the following three functions.

(1) Receiver Buffer Register (RBR): Read-only register for buffering received data.
The RBR register is a data register for storing 5 bits to 8 bits of data, depending on character length.
The bit 0 of the data word is always the first serial data bit that is transmitted or received. When the UART carries out parallel-to-serial or serial-to-parallel conversion operation, the ACE data register has the double buffer configuration so that read operations can be made.
For the UARTRBR register, only read operations can performed with the program when LCR[7] = 0.
The reset value is undefined.

(2) Transmitter Holding Register (THR): Write-only register for setting transmitted data.
The UARTTHR register is a data register for storing 5 bits to 8 bits of data, depending on character length.
When data of less than 8 bits is transmitted, the data is right-aligned to the LSB. The bit 0 of the data word is always the first serial data bit that is transmitted. When the UART carries out parallel-to-serial or serial-to-parallel conversion operation, the ACE data register has the double buffer configuration so that write operations can be made.
For the UARTTHR register, only write operations can be performed with the program when LCR[7] = 0.

(3) Divisor Latch LSB (DLL): 16-bit divisor latch (LSB) for the baud rate generator.
DLL register read/write operations can be performed with the program when LCR[7] = 1. For details, see “Baud rate clock generation”.

13.2.3 UART Interrupt Enable Register / UART Baud Rate Dividing Register(MSB) (UARTIER, UARTDLM)

Address: 0x4008_1004
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	IER3	IER2	IER1	IER0
Access	-	-	-	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
UARTIER is a special function register (SFR) that provides the following two functions.

(1) Interrupt Enable Register (IER): Register used to enable interrupts. IER is used for independently enabling the four serial communication channel interrupts that make the interrupt active.
UARTIER register read/write operations can be performed with the program, when LCR[7] = 0.

[Description of Bits]

- **IER0** (bit 0)
Enables/disables Received Data Available interrupt (in FIFO mode, this includes character timeout interrupt).

IER0	Description
0	Disables Received Data Available interrupt (in FIFO mode, this includes character timeout interrupt)
1	Enables Received Data Available interrupt (in FIFO mode, this includes character timeout interrupt)

- **IER1** (bit 1)
Enables/disables Transmitter Holding Register Empty interrupt.

IER1	Description
0	Disables Transmitter Holding Register Empty interrupt
1	Enables Transmitter Holding Register Empty interrupt

- **IER2** (bit 2)
Enables/disables Receiver Line Status interrupt.

IER2	Description
0	Disables Receiver Line Status interrupt
1	Enables Receiver Line Status interrupt

- **IER3** (bit 3)
Reserved bit. Always set to "0".

(2) Divisor Latch MSB (DLM): 16-bit divisor latch (MSB) for the baud rate generator.
DLM register read/write operations can be performed with the program, when LCR[7] = 1. For details, see “Baud rate clock generation”.

13.2.4 UART Interrupt Status Register / UART FIFO Control Register (UARTIIR, UARTFCR)

Address: 0x4008_1008
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	IIR7	IIR6	-*	-*	IIR3	IIR2	IIR1	IIR0
Access	-	-	-	-	-	-	-	-	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
UARTIIR is a special function register (SFR) that provides the following two functions.
This register will access other registers for read/write operations. Therefore, read-modify-write is prohibited.

(1) Interrupt Identification Register (IIR): Read-only register for interrupt information.
The UARTIIR register stores information indicating that an interrupt with a certain priority level is pending, along with the type of that interrupt. UARTIIR indicates the interrupt with the highest priority level that is pending. All other interrupts will not be recognized until the CPU clears the interrupt. For the UARTIIR register, only read operations can be performed with the program.

[Description of Bits]

- **IIR0** (bit 0)
  Indicates whether an interrupt was generated.

IIR0	Description
0	Interrupt was generated
1	Interrupt was not generated

- **IIR3-1** (bit 3-1)
  Indicates the interrupt sources.

IIR3-1	LVL	Flag	Soruce	Reset Process
011	1	Receiver Line Status	OverrunError/ ParityError/ FramingError/ BreakInterrupt	Read LSR
010	2	Received Data Available	16450 compatible mode: receive data available. FIFO mode: trigger level has been reached.	Read RBR, or when FIFO drops below trigger level
110	2	Character Timeout Indication	At least one character is present in the receive FIFO, and no other character was placed into or read out within 4 character time.	Read RBR
001	3	Transmitter Holding Register Empty (THRE)	16450 compatible mode: THR write enabled. FIFO mode: transmit FIFO is empty.	Read IIR or write THR

- **IIR7-6** (bit **7-6**)
  Indicates operation in FIFO mode.

IIR7-6	Description
00	Non-FIFO mode
01	Unused
10	Unused
11	FIFO mode

(2) FIFO Control Register (FCR): Write-only register for FIFO settings.
The FCR register is used for enabling and clearing the FIFO. The trigger level of the receive FIFO is also set with this register.
For the FCR register, only write operations can be performed with the program.

- **FCR0 (IIR0)** (bit 0)
  FIFO Enable bit. Enables/disables the FIFO.

FCR0	Description
0	Disables FIFO (16450 compatible mode)
1	Enables FIFO

[Note]
FIFO will be cleared when you switch between FIFO enable/disable.

- **FCR1 (IIR1)** (bit 1)
  RCVR FIFO reset.Clears the receive FIFO.

FCR1	Description
0	Normal operation
1	Clears the receive FIFO

[Note]
This bit does not clear the receive shift register.

- **FCR2 (IIR2)** (bit 2)
  XMIT FIFO reset.Clears the transmit FIFO.

FCR2	Description
0	Normal operation
1	Clears the transmit FIFO

[Note]
This bit does not clear the transmit shift register.

- **FCR7-6 (IIR7-6)** (bit **7-6**)
  RCVR FIFO Interrupt Trigger Level bit. Sets the trigger level of the receive FIFO interrupt as follows:

FCR7-6	Description
00	1 byte
01	4 bytes
10	8 byte
11	14 bytes

13.2.5 UART Line Control Register (UARTLCR)

Address: 0x4008_100C
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	LCR7	LCR6	LCR5	LCR4	LCR3	LCR2	LCR1	LCR0
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
UARTLCR is a special function register (SFR) used to control the data character format.
The UARTLCR register is used for enabling and clearing the FIFO. The trigger level of the receive FIFO is also set with this register.
For the UARTLCR, only write operations can be performed with the program.

[Description of Bits]

- **LCR1-0** (bit 1-0)
  LCR1 and LCR0 are used to specify the character length.

LCR1	LCR0	Description
0	0	5-bit length (initial value)
0	1	6-bit length
1	0	7-bit length
1	1	8-bit length

- **LCR2** (bit 2)
  LCR2 is used to specify the number of stop bits for the character to be transmitted.

LCR2	Description
0	1 stop bit
1	1.5 stop bits (when character length = 5 bits) 2 stop bits (when character length = 6, 7, or 8 bits)

- **LCR3** (bit 3)
  LCR3 is used to enable/disable parity.

LCR3	Description
0	Disables parity
1	Enables parity

- **LCR4** (bit 4)
  Selects even or odd parity. This is enabled when LCR3 = “1”.

LCR4	Description
0	Odd parity
1	Even parity

- **LCR5** (bit 5)
  Stick Parity bit. When parity is enabled (LCR[3] = “1”), then parity bit transmission and check will always be logical 1 or 0. When LCR[3] = “1” AND LCR[5] = “1”, then parity bit transmission and check will be logical 0 if LCR[4] = “1”. If LCR[4] = “0”, then parity bit transmission and check will be logical 1.

LCR5	Description
0	Output odd or even parity
1	Always output 0 or 1

LCR5	LCR4	LCR3	Description
0	0	1	Output odd parity
0	1	1	Output even parity
1	0	1	Parity output ”1” is forced
1	1	1	Parity output ”0” is forced

- **LCR6** (bit 6)
  Break Control bit. Sends out a break signal. When LCR[6] = “1”, the serial output (TXD0) is set to a spacing state (0). The break state can be disabled by setting LCR[6] = “0”. The Break Control bit is valid only for TXD0, i.e. TXD0 will be masked but the transmit operation will continue internally. Break Control allows the CPU to send an alarm to the terminal of the computer communication system.

LCR6	Description
0	Normal operation
1	Sends out a break signal

- **LCR7** (bit 7)
  Devisor Latch Access Bit (DLAB). Selects whether to access UARTDLL/UARTDLM or to access UARTRBR/UARTTHR/UARTIER.

LCR7	Description
0	Normal operation (allows access to UARTRBR, UARTTHR, UARTIER)
1	Divisor Latch access (allows access to UARTDLL, UARTDLM)

13.2.6 UART Line Status Register (UARTLSR)

Address: 0x4008_1014
Access: R
Access size: 32 Bits
Initial value: 0x0000_0060

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	LSR7	LSR6	LSR5	LSR4	LSR3	LSR2	LSR1	LSR0
Access	-	-	-	-	-	-	-	-	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
UARTLSR is a special function register (SFR) used to display the status.
UARTLSR is normally the first register read out by the CPU for determining the interrupt cause or for polling the status of the serial communication channel.
When any one of the conditions LSR[1] to LSR[4] is detected, it is an error condition for the Receiver Line Status to generate an interrupt [interrupt of parity 1 in the Interrupt Identification Register (URATIIR)]. This interrupt is enabled by setting IER[2] = “1” in the UARTIER register.

[Description of Bits]

- **LSR0** (bit 0)
  Data Ready bit. This bit is set to 1 when the input character has been received and transferred to the UARTRBR register. This bit is cleared when the UARTRBR register data is read out.

LSR0	Description
0	No valid data in the RBR register
1	Valid data present in the RBR register

- **LSR1** (bit 1)
  Indicates that an overrun error occurred. Overrun error bit indicates that the CPU did not read the data in the UARTRBR register before the next character was sent to the UARTRBR register and overwrote the previous character. In FIFO mode, an overrun error occurs after the next character has been completely received when the FIFO is full. UARTLSR register read operation performed after an overrun error will clear the overrun error. Although the character in the shift register is not transferred to the FIFO, it will be overwritten. This bit is cleared when the UARTLSR register data is read out.

LSR1	Description
0	No overrun error
1	Overrun error occurred

- **LSR2** (bit 2)
  Indicates that a parity error occurred. This is enabled only when parity is enabled. This bit is cleared when the UARTLSR register data is read out. Also, in FIFO mode, this bit indicates that the error is associated with the leading data in the FIFO. Even if a parity error associated with any data other than the leading data in the FIFO had occurred, it will not be indicated by LSR[2].

LSR2	Description
0	No parity error
1	Parity error occurred

- **LSR3** (bit 3)
  Indicates that a framing error occurred. A framing error indicates that there is no valid stop bit in the received character. This bit is set to "1" when the stop bit after the last data bit or after the parity bit is "0" (spacing level). This bit will be cleared when the CPU reads out the UARTLSR register. In FIFO mode, the framing error is related to a specific character in the FIFO. LSR[3] indicates that an error is present when that character comes to the beginning of the FIFO.

LSR3	Description
0	No framing error
1	Framing error occurred

- **LSR4** (bit 4)
  Indicates that a break interrupt occurred. This bit is set to "1" when the input data is maintained in the spacing ("0") state during the transmission of one frame (start bit + data bit + parity bit + stop bit). This bit will be cleared when the CPU reads out the UARTLSR register. In FIFO mode, this is related to a specific character in the FIFO. This bit reflects the break interrupt state when the break character comes to the beginning of the FIFO. If the related character comes to the beginning of the FIFO before the first UARTLSR register is read, the CPU erases this error. When a break interrupt occurs, only one zero character will be loaded into the FIFO.

LSR4	Description
0	No break interrupt
1	Break interrupt occurred

- **LSR5** (bit 5)
  Transmitter Holding Register Empty (THRE).This bit indicates that preparations have been made for calling a new character to be transmitted by the ACE. This bit is set to "1" when the character in the UARTTHR register is transferred to the Transmitter Shift Register. UARTTHR register write operation will clear this bit to "0". This bit will not be cleared by reading out the UARTLSR register. In FIFO mode, this bit is set when the transmit FIFO is empty. This bit is cleared when one byte is written to the transmit FIFO. When THRE interrupt has been enabled by IER1, THRE generates an interrupt in the UARTIIR register with an interrupt priority level 3. When THRE is the interrupt source indicated in the UARTIIR register, this bit will be cleared by reading out the UARTIIR register.

LSR5	Description
0	Transmit data still present in the THR
1	THR ready for transmission

- **LSR6** (bit 6)
  Transmitter Empty.This bit is set to “1” when both the UARTTHR register and the Transmitter Shift Register (TSR) are empty. The UARTLSR register will be cleared to “0” when a character is loaded into the UARTTHR register, and this “0” state is maintained until that character is transferred out from TXD0. This bit will not be cleared to “0” by reading out the UARTLSR register. In FIFO mode, this bit is set to “1” when both the transmit FIFO and the shift register are empty.

LSR6	Description
0	Transmit data still present in either THR or TSR register
1	THR and TSR registers are both empty

- **LSR7** (bit 7)
  This bit is always “0” in the 16450 compatible mode. In FIFO mode, this bit is set to “1” when at least one Parity Error, Framing Error, or Break Interrupt is present in the data within the FIFO. This bit will be cleared when the data causing the error is read out from the RBR, or when the data causing the error is cleared by first clearing the FIFO and then reading out the LSR.

LSR7	Description
0	No data error in FIFO mode
1	A parity error, framing error, or break interrupt occurred in FIFO mode

13.2.7 UART Scratchpad Register (UARTSCR)

Address: 0x4008_101C
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	SCR7	SCR6	SCR5	SCR4	SCR3	SCR2	SCR1	SCR0
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
UARTSCR is a special function register (SFR) intended for storing data temporarily. This register does not affect ACE's transmit and receive operations.

13.3 Description of Operation

The ACE's serial communication channel is programmed by the control registers UARTLCR, UARTIER, UARTDLL, and UARTDLM. These control words define the character length, number of stop bits, parity, baud rate, etc.
Although the order of writing the control registers is immaterial, since UARTIER controls the interrupt enabling, it is necessary to write the UARTIER register in the end. Once the serial communication channel is programmed and becomes ready to operate, these registers can be updated at any time when data is not being transmitted or received.

13.3.1 Data Transmission

Figure 13-2 shows the data transmission timing.

Writing data to the UARTTHR register will transfer the contents through the transmit FIFO to the transmit shift register. Within 16 baud rate clocks after the THRE bit rise is detected, the start bit is sent, followed by the data one bit at a time from the least significant bit. When the data to be transmitted is 7-bit, the most significant bit will not be sent.
If parity is enabled with the LCR3 bit of the UARTLCR register, then the parity bit is sent. This is followed by the stop bit which indicates the end of transmitting one frame of data.
After the data is transmitted, the LSR5 bit of the UARTLSR register is set to "1" to indicate that it is ready for the next transmission. This bit is cleared when one byte is written to the transmit FIFO. Also, when the THRE interrupt has been enabled by IER1, THRE generates an interrupt in the UARTIIR register with an interrupt priority level 3. When THRE is the interrupt source indicated in the UARTIIR register, this bit will be cleared by reading out the UARTIIR register.

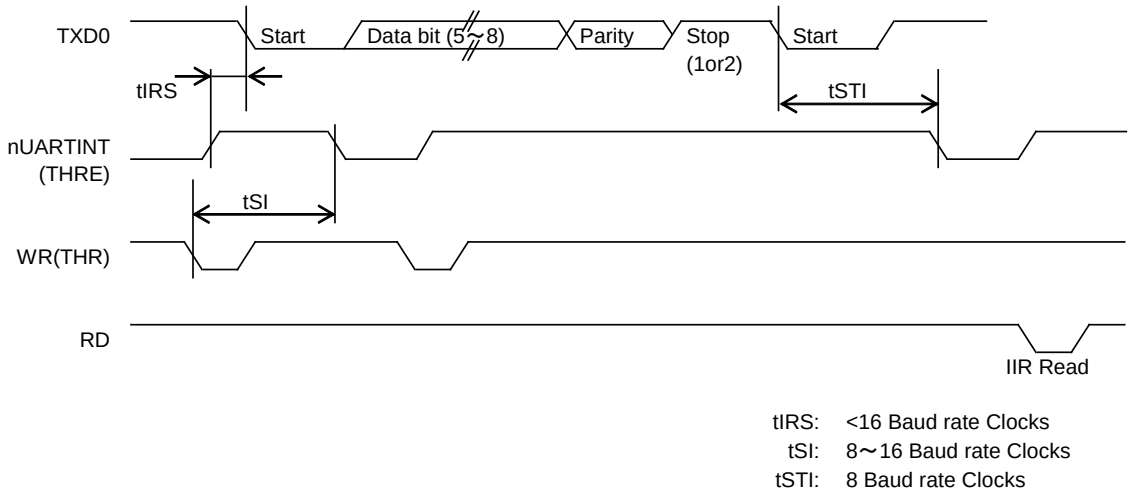


Figure 13-2 Transmission Timing

13.3.2 Data Reception

Figure 13-4 shows the data reception timing. Figure 13-5 shows the reception timing when the first byte in the receive FIFO is read out, and Figure 13-6 shows the reception timing when the remaining bytes in the receive FIFO are read out.

The sampling clock is obtained by dividing the baud rate clock by 1/8.
First, when the start bit is detected from RXD0, subsequent data is obtained and transferred to the receive shift register. The data in the receive shift register is passed to the receive FIFO to be transferred to the UARTRBR register.
When the data reaches the UARTRBR register, LSR0 bit of the UARTLSR register is set to "1" to indicate that valid data is present in the UARTRBR register. This bit will be cleared by reading out the UARTRBR register data.

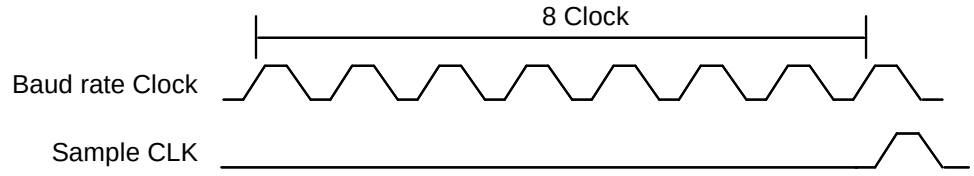


Figure 13-3 Relation between the Baud Rate Clock and Sample Clock

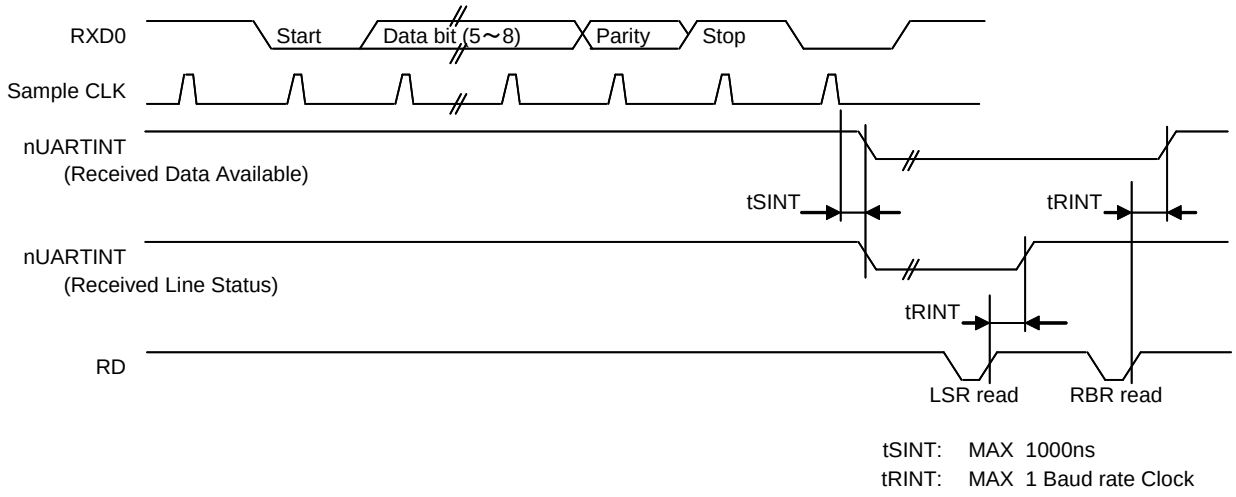


Figure 13-4 Reception Timing

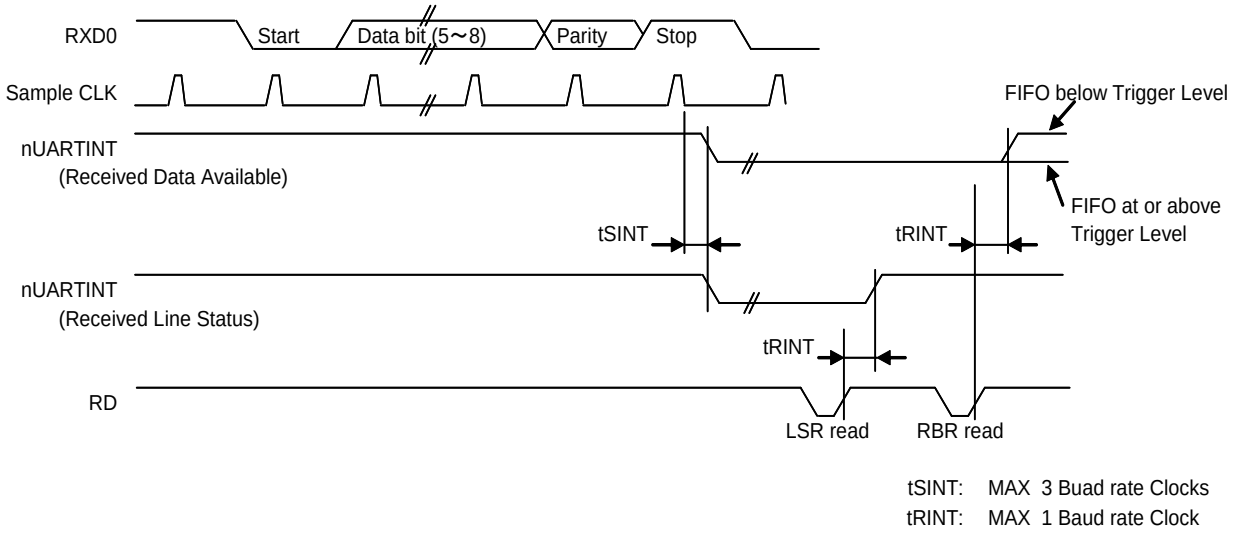


Figure 13-5 First Byte of the Receive FIFO (Set RDR)

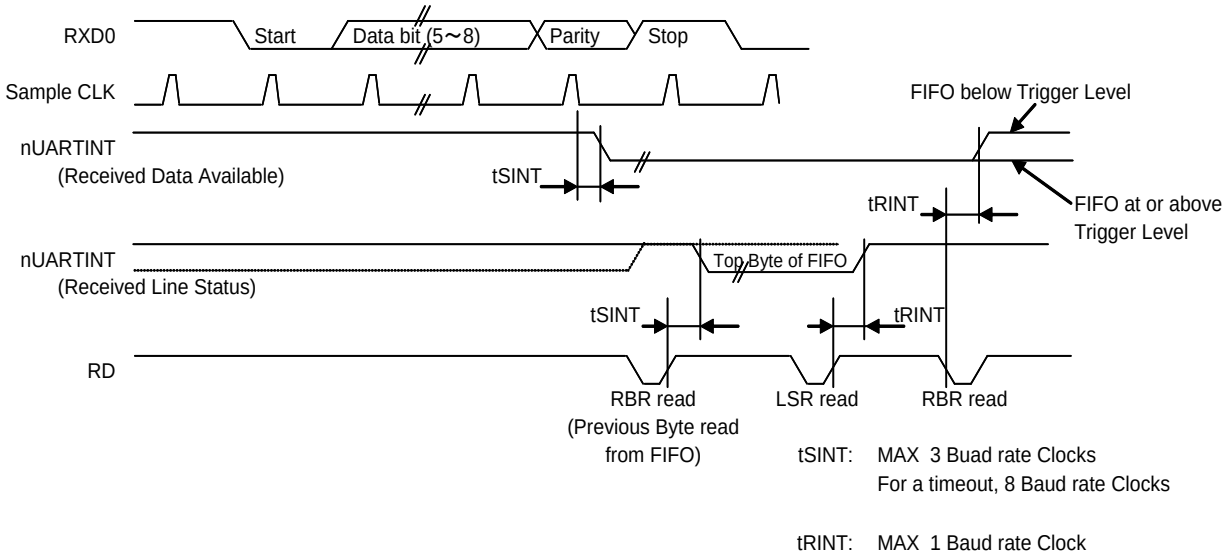


Figure 13-6 Remaining Bytes in the Receive FIFO

13.3.3 Baud Rate Clock Generation

A baud rate is obtained by the following expression:

Baud rate frequency = (SYSCLK * 12 / 13) / (DL[15:0] * 16)

The actual baud rate available for communication will depend on the software process. Under ideal conditions, setting DL = 2 should enable communication. Make sure that the margin of error between the actual and set baud rates is within a few percent.

[Note]
Divisor (DL[15:0]) value cannot be set to 1. Set a value of 0 (stop) or greater than 2.

The following table shows the relation among SYSCLK, DL, and baud rates.

Baud rate (bps)	SYSCLK=32MHz	
	DL(Hex)	Error (%)
300	180A	-0.002
600	0C05	-0.002
1200	0602	0.030
2400	0301	0.030
4800	0181	-0.100
9600	00C0	0.160
19200	0060	0.160
38400	0030	0.160
57600	0020	0.160
115200	0010	0.160

13.3.4 FIFO Mode

When the receive FIFO and reception interrupt are both enabled, reception interrupts are generated as follows:

(A) If the number of characters present within the FIFO exceeds the programmed trigger level, a Received Data Available interrupt is generated. This interrupt is immediately cleared when the number of characters present within the FIFO drops below the trigger level.
(B) As with the Received Data Available interrupt, the Received Data Available flag of the UARTIIR register will be set if the number of characters present within the FIFO exceeds the trigger level, and cleared when the number of characters drops below the trigger level.
(C) The Receiver Line Status interrupt has a higher priority level than the Received Data Available interrupt.
(D) The Received Data Available flag is set as soon as the data in the receive shift register is transferred to the FIFO, and cleared when the FIFO becomes empty.

When the receive FIFO and reception interrupt are both enabled, Character Timeout interrupts are generated as follows:

(A) The Character Timeout interrupt is generated when the following conditions are met.
- There is at least one character present in the FIFO.
- An amount of time required to transfer at least 4 characters has elapsed since a character was last received (if 2 stop bits are specified, the time after the first stop bit is calculated).
- An amount of time required to transfer at least 4 characters has elapsed since the receive FIFO was last read.

For example, if 1 start bit + 8 character bits + 1 parity bit + 2 stop bits is specified, and the transfer speed is 300 baud, the said amount of time will be approximately 160 ms.

(B) The clock used to calculate the character time is CCLK.
(C) When a character is read out from the FIFO, the Character Timeout interrupt and timer used for timeout detection will be cleared.
(D) When no Character Timeout interrupt is generated, the timeout detection timer will be cleared when a character is read out from the FIFO or a new character is received.

The transmission interrupt is generated as follows when the transmit FIFO interrupts have been enabled.

(A) The Transmitter Holding Register Empty interrupt is generated when the transmit FIFO is empty. This interrupt is cleared when a character is written to the transmit FIFO or when UARTIIR is read out.
(B) When the following conditions are met, the Transmitter Holding Register Empty interrupt will be delayed for an amount of time equivalent to “time required to transmit one character – time when last stop bit occurred”.
- There was a point in time where only one character was present in the FIFO after the THRE (Transmitter Holding Register Empty) was last set.
- THRE was set.

13.3.5 FIFO Polled Mode

When FIFO is enabled AND IER[3:0] is all set “0”, the UART will operate in FIFO polled mode. Since the receiver section and transmitter section can be controlled separately, either one (or both) can be set to FIFO polled mode. In FIFO polled mode, the states of the receiver and transmitter sections must be checked by reading out the LSR (since no interrupt is generated).

- A state in which at least one character is present in the receive FIFO can be confirmed by the value “1” set to LSR[0].
- When IER[2] is cleared to “0”, an interrupt will not be generated even if an error is detected while receiving a character. The error state will not be indicated on the IIR value. Therefore, the error type must be checked with the values for LSR[4:1].
- A state in which the transmit FIFO is empty can be confirmed by the value “1” set to LSR[5].
- A state in which the transmit FIFO and transmit shift register are both empty can be confirmed by the value “1” set to LSR[6].
- A state in which the character associated with an error at the time of reception is present in the receive FIFO can be confirmed by the value “1” set to LSR[7].

In FIFO polled mode, FIFO will operate; however, trigger level and timeout detection will not be performed (since they are only notified by interrupts).

13.3.6 Error Status

(a) Overrun error

An overrun error indicates that the data in the UARTRBR register was not read out before the next character was sent to the UARTRBR register and overwrote the previous character.
This error will set the LSR[1] bit of the UARTLSR register.

(b) Parity error

A parity error indicates that the parity of the received data and the received parity bit did not match. This error will set the LSR[2] bit of the UARTLSR register.
Note that, this error will only occur when parity is enabled.
In FIFO mode, this error is associated with the leading data in the FIFO. Even if a parity error associated with any data other than the leading data in the FIFO had occurred, it will not be indicated on the LSR[2] bit of the UARTLSR register.

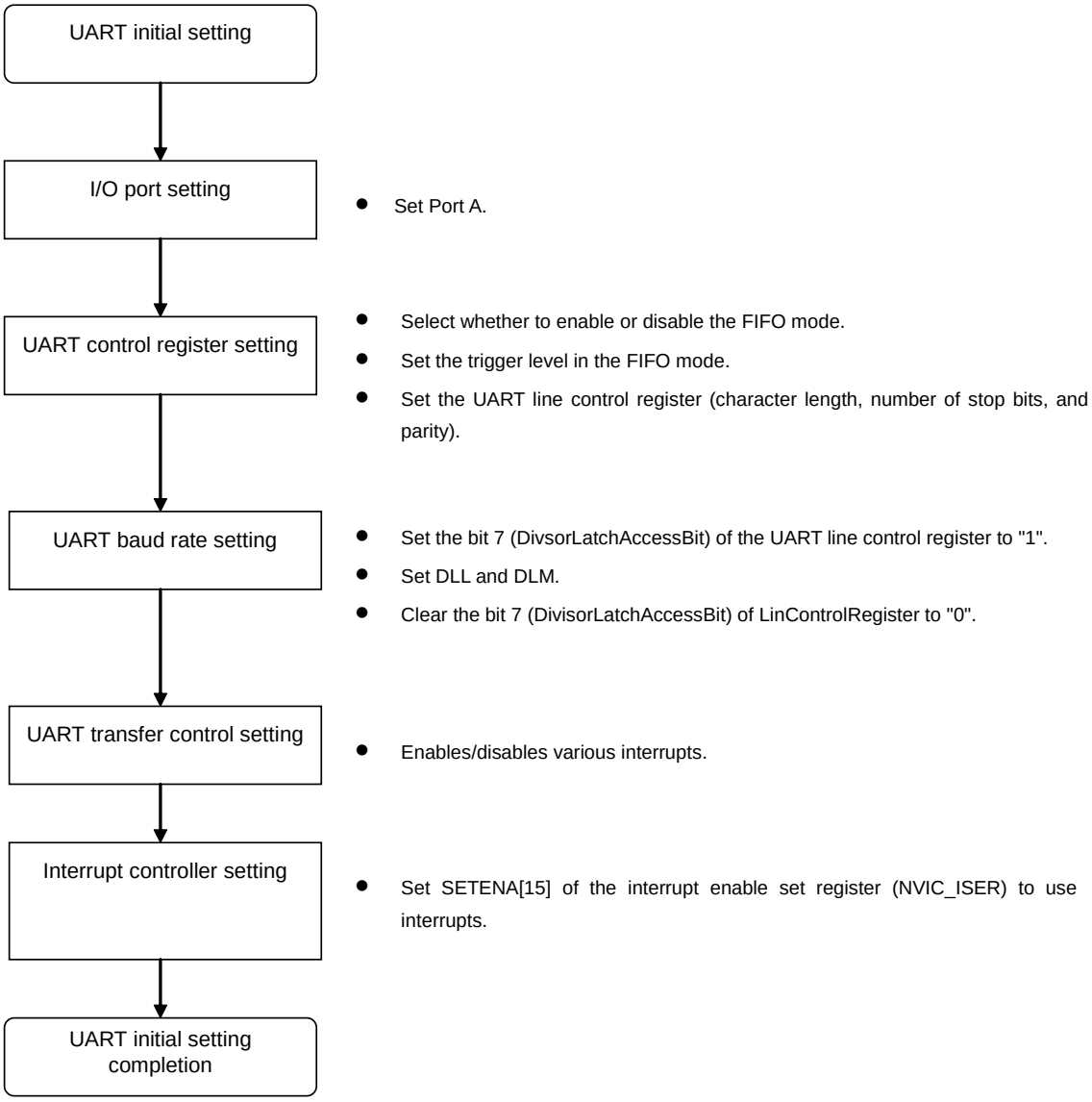
(c) Framing error

A framing error indicates that there is no valid stop bit in the received character. This error will occur when the stop bit after the last data bit or after the parity bit is “0” (spacing level).
This error will set the LSR[3] bit of the UARTLSR register.
In FIFO mode, this is related to a specific character in the FIFO. LSR[3] indicates that an error is present when that character comes to the beginning of the FIFO.

(d) Break interrupt

A break interrupt indicates that the input data was maintained in the spacing (“0”) state during the transmission of one frame (start bit + data bit + parity bit + stop bit).
This error will set the LSR[4] bit of the UARTLSR register.
In FIFO mode, this is related to a specific character in the FIFO. LSR[4] indicates that the break character is present at the beginning of the FIFO.

13.3.7 Setting Example


UART initial setting
I/O port setting
Set Port A.
UART control register setting
Select whether to enable or disable the FIFO mode.
Set the trigger level in the FIFO mode.
Set the UART line control register (character length, number of stop bits, and parity).
UART baud rate setting
Set the bit 7 (DivsorLatchAccessBit) of the UART line control register to "1".
Set DLL and DLM.
Clear the bit 7 (DivisorLatchAccessBit) of LinControlRegister to "0".
UART transfer control setting
Enables/disables various interrupts.
Interrupt controller setting
Set SETENA[15] of the interrupt enable set register (NVIC_ISER) to use interrupts.
UART initial setting completion

13.4 Specifying Port Registers

To use UART, the applicable bit of each related port register needs to be set. See Chapter 15, "GPIO" for details about the port registers.

13.4.1 Operating UART

Set PA2 (bit 9, 8) of the PAMOD register to “01” to select the secondary function (RXD0) of PA2.
Set PA3 (bit 13, 12) of the PAMOD register to “01” to select the secondary function (TXD0) of PA2.

Register name	PAMOD register (address: 0x4000_0260)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	PA3		-	-	PA2		-	-	PA1		-	-	PA0	
Setting value	*	*	**0**	**1**	*	*	**0**	**1**	*	*	*	*	*	*	*	*

Register name	PAMOD register (address: 0x4000_0260)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	PA6		-	-	PA5		-	-	PA4	
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Set the bit 2 of the PIODIR register to “1” to set the input/output mode of PA2 to input.
Set the bit 3 of the PIODIR register to “0” to set the input/output mode of PA3 to output.

Register name	PIODIR register (address: 0x4000_A004)															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-	-	-	-	-	-	-	-	-	-	PIODIR					
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	**0**	**1**	*	*

Register name	PIODIR register (address: 0x4000_A004)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Setting value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

* : Bit not related to the UART function

[Note]
The PIOCON register does not need to be set. When the secondary function is selected, the setting is automatically changed, but the value of the PIOCON register does not change.

Chapter 14 Arithmetic Circuit

14. Arithmetic Circuit

14.1 Overview

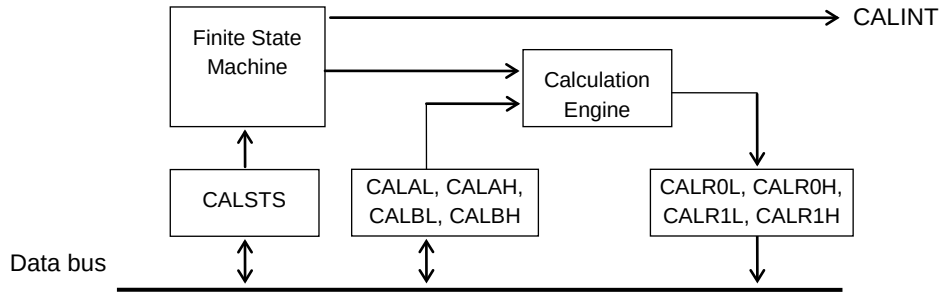
This LSI includes an arithmetic circuit that implements the root operation and division functions.

14.1.1 Features

- Root operation: SQRT(integer (48 bits)) = integer (24 bits), decimal (23 bits) (operation time 48 cycles)
- Division (unsigned): 64 bits/48 bits (operation time 64 cycles)

14.1.2 Configuration

Figure 14-1 shows the configuration of the arithmetic circuit.



CALSTS : Operation status register
CALAL,CALAH, : Operation input register A
CALBL,CALBH, : Operation input register B
CALR0L,CALR0H : Calculation result register 0
CALR1L,CALR1H : Calculation result register 1

Figure 14-1 Configuration of Arithmetic Circuit

14.2 Description of Registers

14.2.1 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4004_5000	Operation status register	CALSTS	R/W	32	0x0000_0000
0x4004_5008	Operation input register AL	CALAL	R/W	32	0x0000_0000
0x4004_500C	Operation input register AH	CALAH	R/W	32	0x0000_0000
0x4004_5010	Operation input register BL	CALBL	R/W	32	0x0000_0000
0x4004_5014	Operation input register BH	CALBH	R/W	32	0x0000_0000
0x4004_5018	Calculation result register 0L	CALR0L	R	32	0x0000_0000
0x4004_501C	Calculation result register 0H	CALR0H	R	32	0x0000_0000
0x4004_5020	Calculation result register 1L	CALR1L	R	32	0x0000_0000
0x4004_5024	Calculation result register 1H	CALR1H	R	32	0x0000_0000

14.2.2 Operation Status Register (CALSTS)

Address: 0x4004_5000
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	SQSTS	–*	–*	DIV_EN
Access	–	–	–	–	–	–	–	–	–	–	–	–	R/W	–	–	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALSTS is a register for the start control and the status confirmation of the operation circuit. Writing "1" to the CALSTS causes the corresponding calculation to begin. The status is maintained as "1" during the calculation, and cleared to "0". At the same time, an interrupt is issued. Stopping a current calculation and starting a next calculation during an operation is prohibited. If it is written, the operation result is not guaranteed.
Setting multiple operations at a time is prohibited.

Description of Bits

- **DIV_EN** (bit 0)
  DIV_EN indicates the start and status of an unsigned division operation.

DIV_EN	Description
0	Division (unsigned) operation stopped (initial value)
1	Division (unsigned) operation proceeding

- **SQSTS** (bit 3)
  SQSTS indicates the start and status of a root operation.

SQSTS	Description
0	Root operation stopped (initial value)
1	Root operation proceeding

14.2.3 Operation Input Register AL(CALAL)

Address: 0x4004_5008
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CALA[31:16]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALA[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALAL is a special function register (SFR) used to store input data for the operation.
CALAL register sets the lower 32 bits of the dividend for division or the lower 32 bits for root operation.
For details, see 14.3, "Description of Operation".

14.2.4 Operation Input Register AH(CALAH)

Address: 0x4004_500C
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CALA[63:48]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALA[47:32]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALAH is a special function register (SFR) used to store input data for the operation.
CALAH register sets the higher 32 bits of the dividend for division or the higher 16 bits for root operation.
For details, see 14.3, "Description of Operation".

14.2.5 Operation Input Register BL(CALBL)

Address: 0x4004_5010
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CALB[31:16]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALB[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALBL is a special function register (SFR) used to store input data for the operation.
CALBL register sets the lower 32 bits of the divisor for division. Setting the divisor to 0 is prohibited.
For details, see 14.3, "Description of Operation".

14.2.6 Operation Input Register BH(CALBH)

Address: 0x4004_5014
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALB[47:32]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALBH is a special function register (SFR) used to store input data for the operation.
CALBL register sets the higher 16 bits of the divisor for division. Setting the divisor to 0 is prohibited.
For details, see 14.3, "Description of Operation".

14.2.7 Calculation Result Register 0L(CALR0L)

Address: 0x4004_5018
Access: R
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CALR0[31:16]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALR0[15:0]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALR0L is a special function register (SFR) that stores the result data of the operation.
CALR0L register indicates the lower 32 bits of the quotient for division, or the decimal part 23 bits and the lower 8 bits of the integer part in the operation result for root operation.
For details, see 14.3, "Description of Operation".

14.2.8 Calculation Result Register 0H(CALR0H)

Address: 0x4004_501C
Access: R
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CALR0[63:48]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALR0[47:32]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALR0H is a special function register (SFR) that stores the result data of the operation.
CALR0H register indicates the higher 32 bits of the quotient for division, or the upper 16 bits of the integer part in the operation result for root operation.
For details, see 14.3, "Description of Operation".

14.2.9 Calculation Result Register 1L(CALR1L)

Address: 0x4004_5020
Access: R
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	CALR1[31:16]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALR1[15:0]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALR1L is a special function register (SFR) that stores the result data of the operation.
CALR1L register indicates the lower 32 bits of the remainder in the division result for division.
For details, see 14.3, "Description of Operation".

14.2.10 Calculation Result Register 1H(CALR1H)

Address: 0x4004_5024
Access: R
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*	–*
Access	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	CALR1[47:32]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CALR1H is a special function register (SFR) that stores the result data of the operation.
CALR1H register indicates the higher 16 bits of the remainder in the division result for division.
For details, see 14.3, "Description of Operation".

14.3 Description of Operation

14.3.1 Division

The formats of the input and result of the division are shown below.

	Input		result	
Integer part/decimal part	Dividend	Divisor	Quotient	Remainder
Division	CALA[63:0]	CALB[47:0]	CALR0[63:0]	CALR1[47:0]

Writing "1" to the DIV_EN of CALSTS causes the calculation to begin. The quotient in the calculation result of CALA[63:0]/CALB[47:0] is stored in CALR0[63:0], and the remainder is stored in CALR1[47:0]. When the operation completes, the set bit is cleared to "0", and an interrupts is issued.

Setting the divisor to 0 is prohibited. Even if it is set to 0, an interrupt occurs after 48 cycles.

14.3.2 Root Operation

The formats of the input and result of the root operation are shown below.

	Input	result	
Integer part/decimal part	Integer part	Integer part	Decimal part
Root operation	CALA[47:0]	CALR0[47:24]	CALR0[23:1]

Writing "1" to the SQSTS of CALSTS causes the root calculation to begin. The integer part in the calculation result of the square root of CALA[47:0] is stored in CALR0[47:24], and the decimal part is stored in CALR0[23:1]. When the operation completes, the SQSTS is cleared to "0", and an interrupts is issued.

Chapter 15 GPIO

15. GPIO

15.1 Overview

7-bit x 1ch general-purpose I/O.
Input or output can be selected for each bit. The interrupt is available with any bit. The interrupt mask and the interrupt mode (level/edge and positive/negative polarity) can be set for all of the bits.

15.1.1 Features

- Input/output selectable for each bit
- Interrupt available with any bit
- The interrupt mask and the interrupt mode (level/edge, positive/negative polarity) can be set for all the bits.
- The sampling mode can be set.

15.1.2 Configuration

Figure 15-1 shows the configuration of the GPIO.

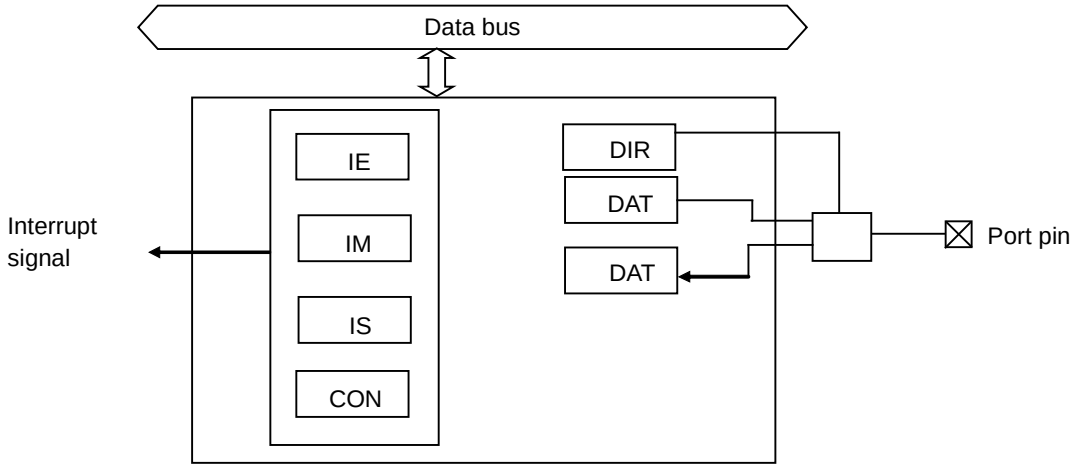


Figure 15-1 Configuration of GPIO

15.1.3 List of Pins

Table 15-1 List of Pins Interfaced with the Outside of LSI

Pin name	I/O	Initial status	Initial value	Description
PA0-6	I/O	I	-	7-bit general-purpose port (GPIOA)

[Note]
For details of switching to the secondary/tertiary functions of the port pin, see Chapter 5, "MCU Control Function".

15.2 Description of Registers

15.2.1 List of Registers

Table 15-2 List of Registers

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4000_A000	GPIO data register	PIODAT	R/W	32	0x0000_0000
0x4000_A004	GPIO direction register	PIODIR	R/W	32	0x0000_0000
0x4000_A008	GPIO port control register	PIOCON	R/W	32	0x0000_0000
0x4000_A010	GPIO interrupt enable register	PIOIE	R/W	32	0x0000_0000
0x4000_A014	GPIO interrupt mode register	PIOIM	R/W	32	0x0000_0000
0x4000_A018	GPIO interrupt status register	PIOIS	R/W	32	0x0000_0000

15.2.2 GPIO Port Data Register (PIODAT)

Address: 0x4000_A000
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	PIODAT[6:0]						
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register is a special function register (SFR) to set the value to be output to the Port A pin or to read the input level of the Port A pin. In output mode, the value of this register is output to the Port A pin. The value written to PIODAT is readable.
In input mode, the input level of the Port A pin is read when PIODAT is read. Output mode or input mode is selected by using the port mode register (PIODIR) described later. Set the value to be output to the port pin. The value of this register is output to the external pin when the value of the port direction register indicates output.

[Description of Bits]
- **PIODAT[6:0]** (bits 6-0)
  PIODAT[6:0] is used to set the output value of the Port A pin in output mode and to read the pin level of the Port A pin in input mode.

PIODAT[n]	Description
0	Output or input level of the PAn pin: ”L”
1	Output or input level of the PAn pin: ”H”

n=0-6

15.2.3 GPIO Port Direction Register (PIODIR)

Address: 0x4000_A004
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	PIODIR[6:0]						
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading.

[Description of Register]
PIODIR is a special function register (SFR) to select the input/output mode of Port A.

[Description of Bits]

- **PIODIR[6:0]** (bits 6-0)
  PIODIR[6:0] is used to set the input/output mode of the Port A pin.

PIODIR[n]	Description
0	PAn pin: Output (initial value)
1	PAn pin: Input

n=0-6

15.2.4 GPIO Port Control Register (PIOCON)

Address: 0x4000_A008
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	PIOCON6 [1:0]		-*	-*	PIOCON5 [1:0]		-*	-*	PIOCON4 [1:0]	
Access	-	-	-	-	-	-	R/W	R/W	-	-	R/W	R/W	-	-	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	PIOCON3 [1:0]		-*	-*	PIOCON2 [1:0]		-*	-*	PIOCON1 [1:0]		-*	-*	PIOCON0 [1:0]	
Access	-	-	R/W	R/W	-	-	R/W	R/W	-	-	R/W	R/W	-	-	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
This register is a special function register (SFR) to select input/output state of the Port A pin. The input/output state is different between input mode and output mode. Input or output is selected by using the PIODIR register.

[Description of Bits]

- **PIOCONn[1:0]** (n=0-6)
  PIOCONn[1:0] (n = 0 - 6) is used to select high-impedance output, P-channel open drain output, N-channel open drain output, or CMOS output in output mode and to select high-impedance input, input with a pull-down resistor, or input with a pull-up resistor in input mode.

Setting of PAn pin	When output mode is selected (PIODIR[n] bit = “0”)	When input mode is selected (PIODIR[n] bit = “1”)
PIOCONn[1:0]	Description	
00	High-impedance output (initial value)	High-impedance input
01	P-channel open drain output	Input with a pull-down resistor
10	N-channel open drain output	Input with a pull-up resistor
11	CMOS output	High-impedance input

n=0-6

In output mode, the output state of the port varies depending on the settings of PIOCONn[1:0] and PIODAT[n], as shown in the table below.

Setting of PAn pin		PIODAT[n]=0	PIODAT[n]=1
PIOCONn[1:0]	Description	PAn pin output state	
00	High-impedance output	Hi-z	Hi-z
01	P-channel open drain output	Hi-z	1
10	N-channel open drain output	0	Hi-z
11	CMOS output	0	1

n=0-6

15.2.5 GPIO Interrupt Enable Register (PIOIE)

Address: 0x4000_A010
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	PIOIE[6:0]						
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
Enables/disables interrupts. However, if a bit is set to the output mode by the PIOPMn register, the bit does not become an interrupt source regardless of the value of the PIOIE register.

[Description of Bits]

- **PIOIE[6:0]** (bits 6-0)

PIOIE[n]	Description
0	Disables PAn pin interrupt
1	Enables PAn pin interrupt

n=0-6

[Note]
The pin state is notified to GPIO even when the secondary function is selected for the pin as an external pin of LSI (the GPIO function is not selected). If the IE bit of a pin set to use the secondary function is set to “1”, an interrupt from GPIO is notified to CPU according to the state of the pin as the secondary function. When the secondary function is used and the interrupt processing from GPIO is not required, set the applicable IE bit to “0”.

15.2.6 GPIO Interrupt Mode Register (PIOIM)

Address: 0x4000_A014
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	PIOIM6[3:0]				PIOIM5[3:0]				PIOIM4[3:0]			
Access	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	PIOIM3[3:0]				PIOIM2[3:0]				PIOIM1[3:0]				PIOIM0[3:0]			
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
Sets the interrupt mode for each GPIO pin.

[Description of Bits]

- **PIOIMn[2:0]**
  Sets the interrupt mode for the PAn (n = 0 - 6) pin.
  An interrupt occurs under the following conditions, according to the PIOIMn setting. (n = 0 - 6)

Value of PIOIMn[2:0]	Interrupt mode
000	An interrupt occurs at a falling edge
001	An interrupt occurs at a rising edge
010	An interrupt occurs with a “L” level input
011	An interrupt occurs with a “H” level input
100	An interrupt occurs at both edges (rising/falling)
Other than above	Setting prohibited

- **PIOIMn[3]**
  Sets whether the PAn (n = 0 - 6) pin interrupt is detected with or without sampling.

Value of PIOIMn[3]	Description
0	Detects without sampling (initial value)
1	Detects with sampling

The detection with sampling performs two cycles of sampling at 16 kHz.

15.2.7 GPIO Interrupt Status Register (PIOIS)

Address: 0x4000_A018
Access: R/W
Access size: 32 Bits
Initial value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	PIOIS[6:0]						
Access	-	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future expansion. "0" is read when reading. Write “0” when writing.

[Description of Register]
Holds the interrupt status. Only writing “1” is valid.
By writing “1” to the bit which caused an interrupt, the interrupt is cleared.

[Description of Bits]

- **PIOIS[6:0]** (bits 6-0)

PIOIS[n]	Description
0	No PAn pin interrupt source exists
1	A PAn pin interrupt source exists

n=0-6

[Note]
If “1” is written to a bit with value 0, the bit value is not changed.

15.3 Description of Operation

15.3.1 I/O Setting

For each pin of Port A, either output or input is selected by setting the GPIO direction register (PIODIR).
In output mode, high-impedance output mode, P-channel open drain output mode, N-channel open drain output mode, or CMOS output mode can be selected by setting the GPIO control registers (PACON).
In the input mode, set the GPIO control registers (PACON) to select any of high-impedance input mode, input mode with a pull-down resistor, or input mode with a pull-up resistor.
At a system reset, high-impedance output mode is selected as the initial status.
In the CMOS output mode, “L” or “H” level is output to each pin of Port A depending on the value set by the GPIO data register (PIODAT).
In input mode, the input level of each pin of Port A is read from the GPIO data register (PIODAT).

15.3.2 Interrupt Setting Procedure

Use the following procedure to set interruption.

①Write “0” to the corresponding bit of the GPIO interrupt enable register (PIOIE) to disable interrupts.
②Set the corresponding bit of the GPIO port direction register (PIODIR) to “1” (input).
③Set the interrupt mode to the corresponding bit of the GPIO interrupt mode register (PIOIM).
④Write “1” to the corresponding bit of the GPIO interrupt status register (PIOIS) to clear the interrupt status.
⑤Write “1” to the corresponding bit of the GPIO interrupt enable register (PIOIE) to enable interrupts.

After the above settings are made, the interrupt is enabled for the corresponding bit.
For the interrupt generation operation, see Chapter 15.3.3, "Various Interrupt Operations".

15.3.3 Various Interrupt Operations

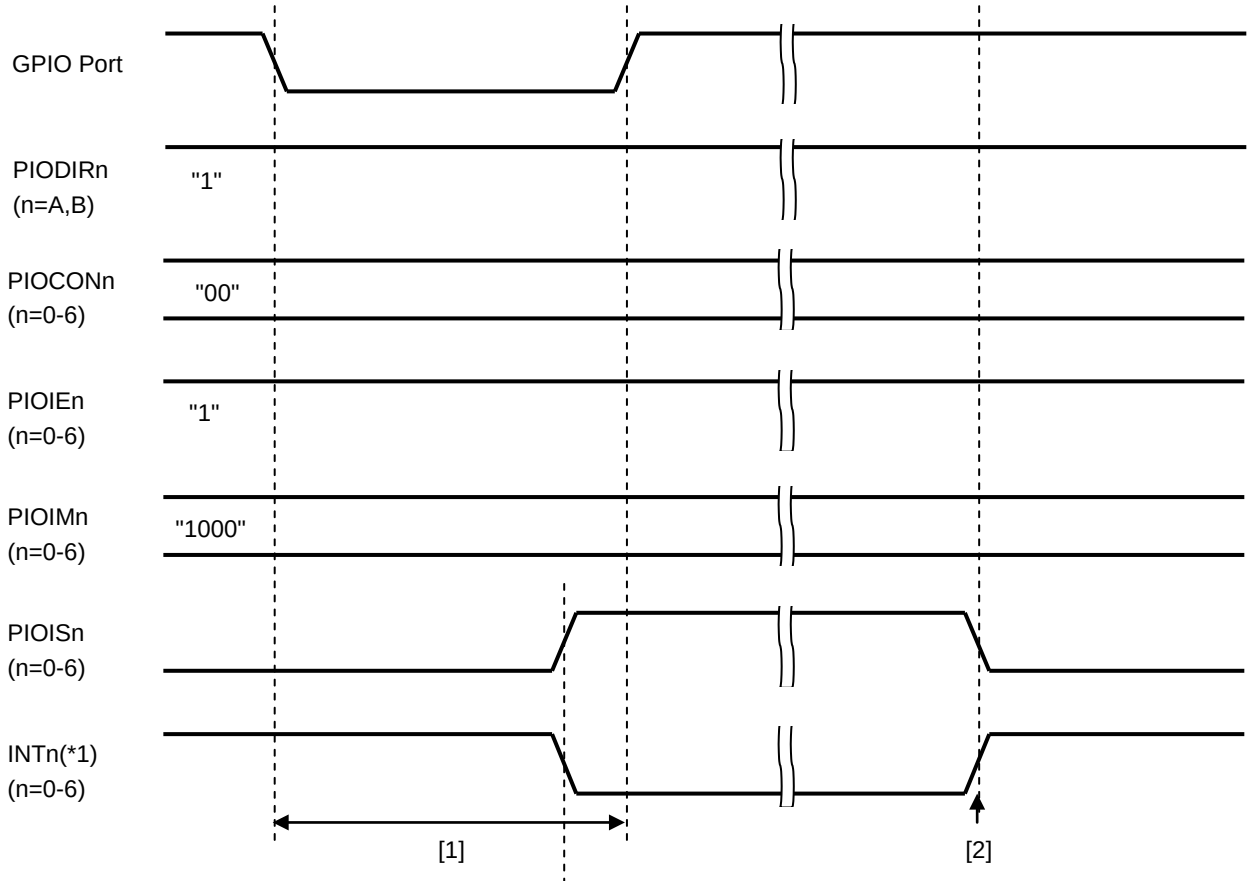
Interrupt processing can be set independently for each bit of each port pin. For an applicable bit, the interrupt can be enabled by setting the GPIO port direction (PIODIR) to “1” and the GPIO interrupt enable register (PIOIE) to “1”. For the interrupt setting procedure, see Chapter 15.3.2, "Interrupt Setting Procedure".

15.3.3.1 Falling Edge Interrupt Mode

●For Detection with Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**1**000", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when a falling edge of the input signal of GPIO pin is detected (Note that the assertion period of "L" after the falling edge must be 125 µs (two cycles at 16 kHz) or longer). By writing "1" to the GPIO interrupt status register (PIOIS) from CPU, the GPIO interrupt status register (PIOIS) is cleared to "0", and the interrupt is released at the same time. If an interrupt source is generated at the same timing as the release of the interrupt from CPU, the interrupt generation has priority.

Figure 15-2 shows an operation example.



[1]Edge interrupt assertion period: At least 125 µs (at least 2 cycles or longer at 16 kHz)

[2]Interrupt status clear (Write "1" to the PIOISn bit)

(*1) Internal interrupt signal

Figure 15-2 Example of Falling Edge Interrupt Operation

●For Detection without Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**0**000", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when a falling edge of the input signal of GPIO pin is detected (Note that the assertion period of "L" after the falling edge must be one cycle of system clock or longer).

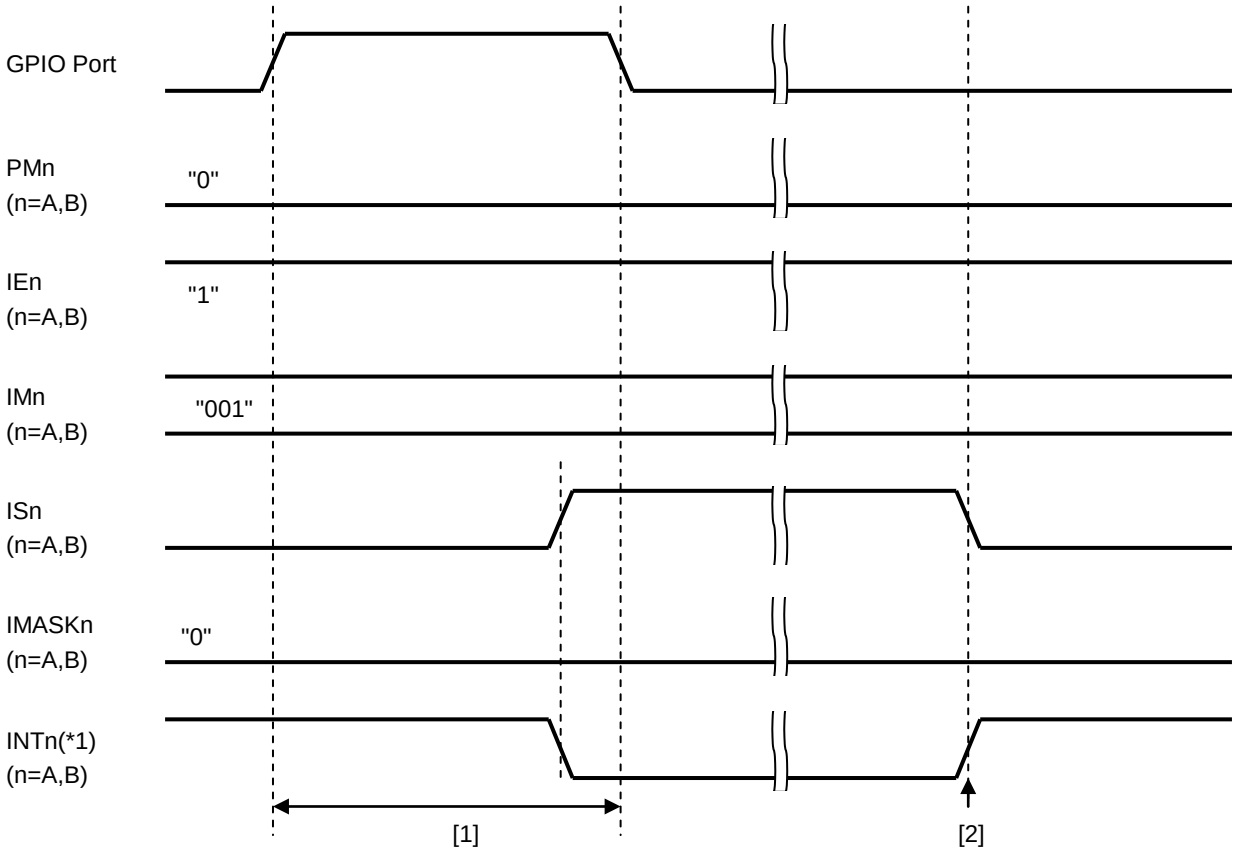
The interrupt is generated after the pin is asserted, or after three sysclk cycles.

* The external input is used after being synchronized with sysclk in two stages. For the detection with sampling, the input signal is synchronized with sysclk through 16 kHz.

15.3.3.2 Rising Edge Interrupt

●For Detection with Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**1**001", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when a rising edge of the input signal of GPIO pin is detected (Note that the width of "H" after the rising edge must be 125 µs (two cycles at 16 kHz) or longer). By writing "1" to the GPIO interrupt status register (PIOIS) from CPU, the GPIO interrupt status register (PIOIS) is cleared to "0", and the interrupt is released at the same time. If an interrupt source is generated at the same timing as the release of the interrupt from CPU, the interrupt generation has priority.
Figure 15-3 shows an operation example.



[1]Edge interrupt assertion period: At least 125 µs (at least 2 cycles or longer at 16 kHz)
[2]Interrupt status clear (Write "1" to the ISn bit)

(*1) Internal interrupt signal

Figure 15-3 Example of Rising Edge Interrupt Operation

●For Detection without Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**0**001", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when a rising edge of the input signal of GPIO pin is detected (Note that the assertion period of "L" after the falling edge must be one cycle of system clock or longer).
The interrupt is generated after the pin is asserted, or after three sysclk cycles.

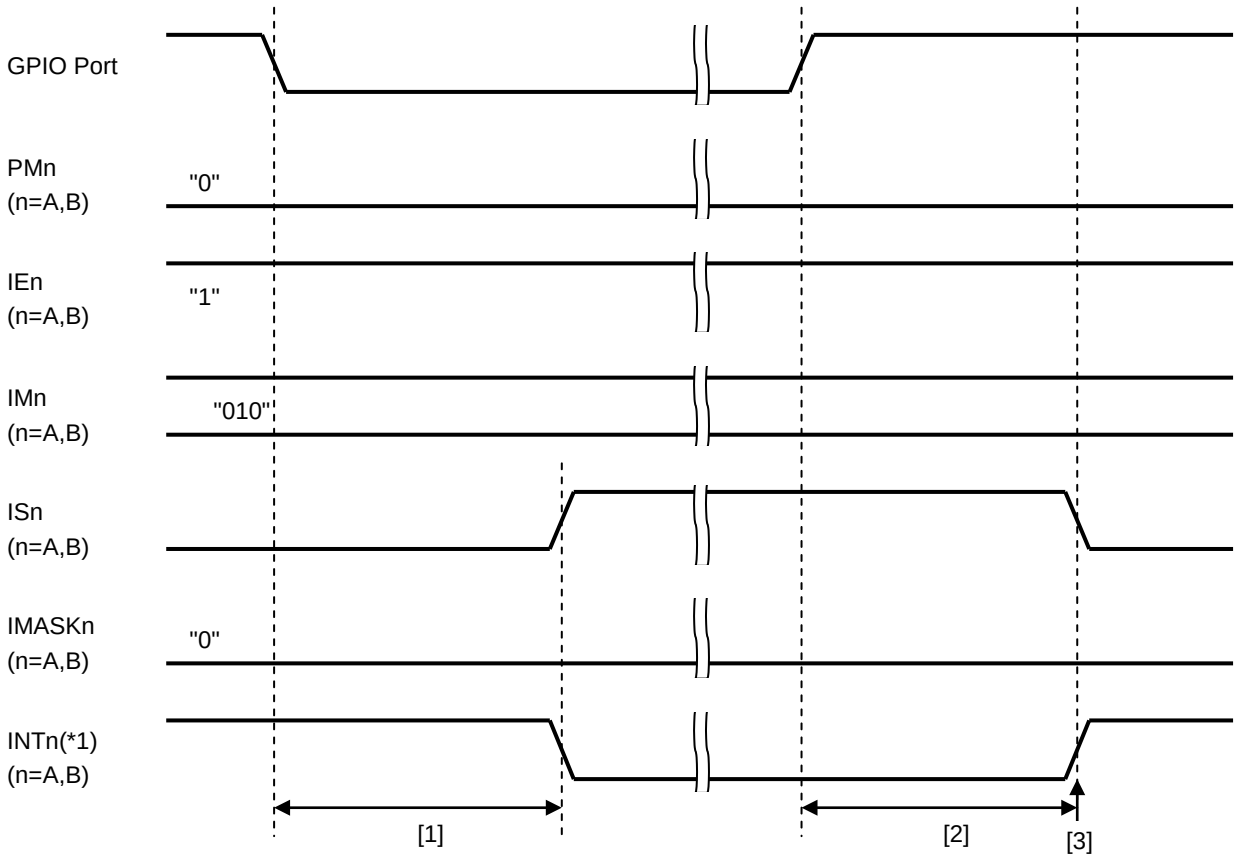
15.3.3.3 L-level Input Interrupt

●For Detection with Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**1**010", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when the input signal of GPIO pin goes into the L level.

By waiting at least 125 µs (two cycles at 16 kHz) after the input signal of GPIO pin goes into the H level and then writing "1" to the GPIO interrupt status register (PIOIS) from CPU, the GPIO interrupt status register (PIOIS) is cleared to "0", and the interrupt is released at the same time. If "1" is written to the GPIO interrupt status register (PIOIS) from CPU while the input signal of GPIO pin is L, the GPIO interrupt status register (PIOIS) is not cleared to "0", and the interrupt is not released. If an interrupt source is generated at the same timing as the release of the interrupt from CPU, the interrupt generation has priority.

Figure 15-4 shows an operation example.



[1]Period from interrupt assertion to internal interrupt generation: Up to 2 cycles at 16 kHz

[2]Interrupt status clear disable period after the interrupt source clear (GPIO pin is deasserted):

Up to 2 cycles at 16 kHz

Even if the interrupt status is cleared during this period, the interrupt signal is not released.

[3]Interrupt status clear (Write "1" to the ISn bit)

(*1) Internal interrupt signal

Figure 15-4 Example of L-level Input Interrupt Operation

●For Detection without Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**0**010", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when the input signal of GPIO pin goes into the L level (Note that the assertion period of "L" must be one system clock cycle or longer).

The interrupt is generated after the pin is asserted, or after three sysclk cycles.

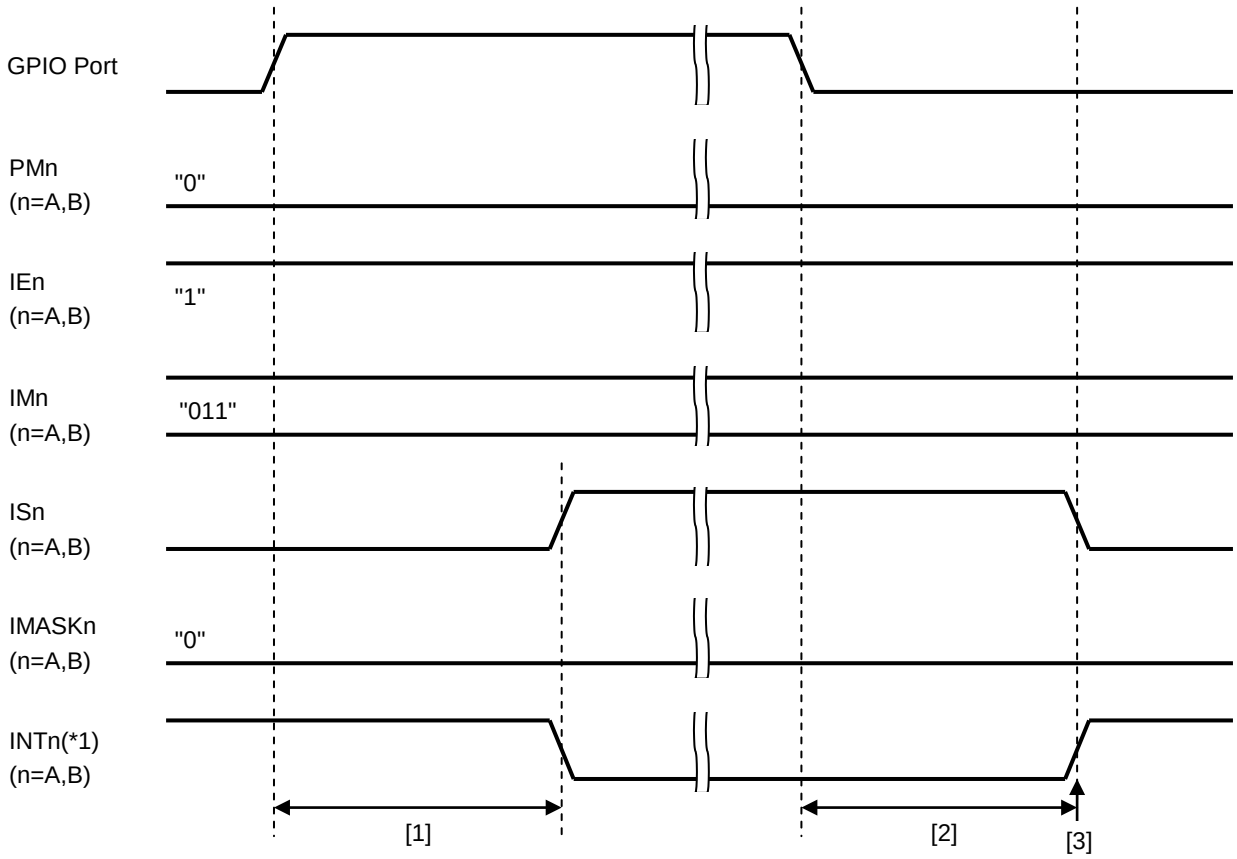
15.3.3.4 H-level Input Interrupt

●For Detection with Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**1**011", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when the input signal of GPIO pin goes into the H level.

By waiting at least 125 µs (two cycles at 16 kHz) after the input signal of GPIO pin goes into the L level and then writing "1" to the GPIO interrupt status register (PIOIS) from CPU, the GPIO interrupt status register (PIOIS) is cleared to "0", and the interrupt is released at the same time. If "1" is written to the GPIO interrupt status register (PIOIS) from CPU while the input signal of GPIO pin is H, the GPIO interrupt status register (PIOIS) is not cleared to "0", and the interrupt is not released. If an interrupt source is generated at the same timing as the release of the interrupt from CPU, the interrupt generation has priority.

Figure 15-5 shows an operation example.



[1]Period from interrupt assertion to internal interrupt generation: Up to 2 cycles at 16 kHz

[2]Interrupt status clear disable period after the interrupt source clear (GPIO pin is deasserted):

Up to 2 cycles at 16 kHz

Even if the interrupt status is cleared during this period, the interrupt signal is not released.

[3]Interrupt status clear (Write "1" to the ISn bit)

(*1) Internal interrupt signal

Figure 15-5 Example of H-level Input Interrupt Operation

●For Detection without Sampling

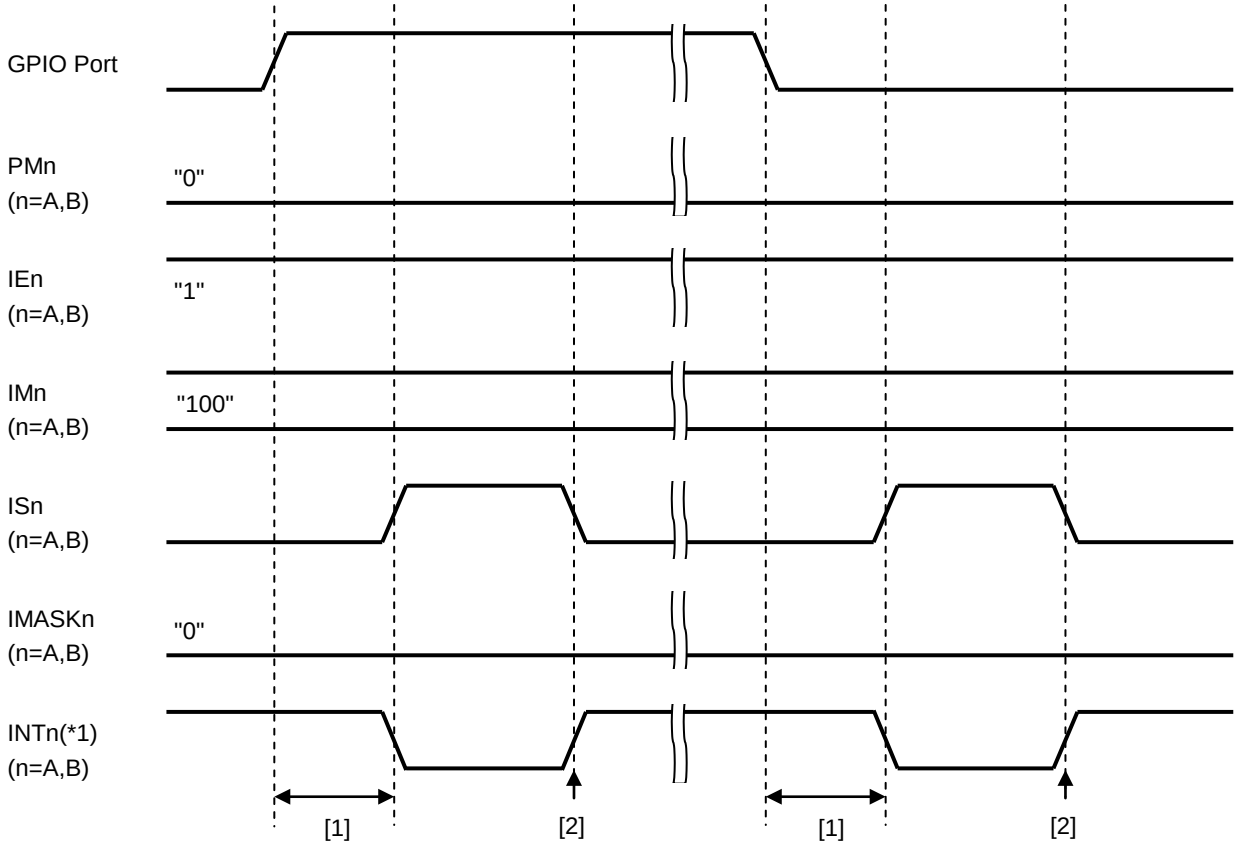
If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**0**011", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when the input signal of GPIO pin goes into the H level (Note that the assertion period of "H" must be one system clock cycle or longer).

The interrupt is generated after the pin is asserted, or after three sysclk cycles.

15.3.3.5 Both-edge (Rising/Falling) Interrupt

●For Detection with Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**1**100", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when a rising or falling edge of the input signal of GPIO pin is detected (Note that the width of "L" after the falling edge and the width of "H" after the rising edge must be 125 µs (two cycles at 16 kHz) or longer). By writing "1" to the GPIO interrupt status register (PIOIS) from CPU, the GPIO interrupt status register (PIOIS) is cleared to "0", and the interrupt is released at the same time. If an interrupt source is generated at the same timing as the release of the interrupt from CPU, the interrupt generation has priority.
Figure 15-6 shows an operation example.



[1]Period from edge interrupt assertion to internal interrupt generation: Up to 2 cycles at 16 kHz
During this period, the status of GPIO pin must be held.
[2]Interrupt status clear (Write "1" to the ISn bit)

(*1) Internal interrupt signal

Figure 15-6 Example of Both-edge (Rising/Falling) Interrupt Operation

●For Detection without Sampling

If the interrupt function is enabled and the GPIO interrupt mode register (PIOIM) is set to "**0**100", the GPIO interrupt status register (PIOIS) is set to "1" to generate an interrupt when a falling edge of the input signal of GPIO pin is detected (Note that the assertion period of "L" after the falling edge and that of "H" after the rising edge must be one cycle of system clock or longer).
The interrupt is generated after the pin is asserted, or after three sysclk cycles.

Chapter 16 Flash Programming

16. Flash Programming

16.1 General Description

This LSI includes the self-rewrite function that rewrites the content of the flash memory (program memory space) using a special function register (SFR) programmatically.

16.1.1 Features

The self-rewrite function of the flash memory has the following features:

- Supports the writing by word (32 bits)
- Supports the erase by sector (128 words)
- Guarantees 1000 rewritings
- Supports the software re-mapping and the hardware re-mapping function with the BRMP pin.

16.2 Description of Registers

16.2.1 List of Registers

Address	Name	Symbol (Word)	R/W	Size	Initial value
0x4000_0400	Flash-ROM status register	FLCSTA	R	32	0x0000_0000
0x4000_0404	Flash-ROM acceptor register	FLCACP	W	32	0x0000_0000
0x4000_0408	Flash-ROM address register	FLCADR	R/W	32	0x0000_0000
0x4000_040C	Flash-ROM write data register	FLCWDA	W	32	0x0000_0000
0x4000_0410	Flash-ROM erase register	FLCERA	R/W	32	0x0000_0000
0x4000_0420	Flash-ROM size register	FLCRSIZ	R	32	0x0002_0000
0x4000_0424	Boot program address register	FLCBADR	R	32	0x0001_F000
0x4000_0460	Set count acceptor register	FLCSCACP	W	32	0x0000_0000
0x4000_046C	Set count sector erase CE enable register	FLCSCSCE	R/W	32	0x0002_1474
0x4000_0470	Set count program CE enable register	FLCSCPCE	R/W	32	0x0000_04FB
0x4000_047C	Set count sector erase WE enable register	FLCSCSWE	R/W	32	0x0002_13AA
0x4000_0480	Set count program WE enable register	FLCSCPWE	R/W	32	0x0000_0431
0x4000_0484	Set count program PROG2_1st enable register	FLCSCPP21E	R/W	32	0x0000_03DD
0x4000_0488	Set count program PROG2_1st disable register	FLCSCPP21D	R/W	32	0x0000_0313
0x4000_048C	Set count program PROG2_2nd enable register	FLCSCPP22E	R/W	32	0x0000_0311
0x4000_0490	Set count program PROG2_2nd disable register	FLCSCPP22D	R/W	32	0x0000_0247
0x4000_0494	Set count program PROG2_3rd enable register	FLCSCPP23E	R/W	32	0x0000_0245
0x4000_0498	Set count program PROG2_3rd disable register	FLCSCPP23D	R/W	32	0x0000_017B
0x4000_049C	Set count program PROG2_4th enable register	FLCSCPP24E	R/W	32	0x0000_0179
0x4000_04A0	Set count program PROG2_4th disable register	FLCSCPP24D	R/W	32	0x0000_00AF
0x4000_04A8	Set count program BYTE1 enable register	FLCSCPB1E	R/W	32	0x0000_0312
0x4000_04AC	Set count program BYTE2 enable register	FLCSCPB2E	R/W	32	0x0000_0246
0x4000_04B0	Set count program BYTE3 enable register	FLCSCPB3E	R/W	32	0x0000_017A
0x4000_04B4	Set count erase WE disable register	FLCSCEWED	R/W	32	0x0000_06AA
0x4000_04B8	Set count program WE disable register	FLCSCPWED	R/W	32	0x0000_00AE
0x4000_04BC	Set count CE disable register	FLCSCCED	R/W	32	0x0000_0006
0x4000_04C0	Set count termination register	FLCSCEND	R/W	32	0x0000_0001

16.2.2 Flash-ROM Status Register (FLCSTA)

Address: 0x4000_0400
Access: R
Access size: 32 Bits
Initial Value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	BUSY
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSTA is a read-only, special function register (SFR) to indicate a state of the Flash-ROM.

[Description of Bits]

- **BUSY** (bit 0)
  Indicates a state of the Flash-ROM controller. This bit is "1" during sector erase/1-word write. It automatically changes to "0" when sector erase/1-word write is completed.

BUSY	Description
0	Sector erase/1-word write is completed
1	During sector erase/1-word write

16.2.3 Flash-ROM Acceptor Register (FLCACP)

Address: 0x4000_0404
Access: W
Access size: 32 Bits
Initial Value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FAC[7:0]							
Access	-	-	-	-	-	-	-	-	W	W	W	W	W	W	W	W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
The read data of this register is "0".
*: Reserved bit for future extension. Write "0" when writing.

[Description of Register]
FLCACP is a write-only special function register (SFR) to control enabling/disabling sector erase and 1-word write operation for Flash-ROM rewrite.

[Description of Bits]

- **FAC[7:0]** (bit 7-0)

FAC[7:0] is a register used to restrict sector erase, block erase, and 1-word write operations in order to prevent an unintended operation.
Writing of "0x0000_00FA" and "0x0000_00F5" to FLCACP in this order enables a one-time sector erase or 1-word write. When you use sector erase or 1-word write in succession, you must write "0x0000_00FA" and "0x0000_00F5" in FLCACP every time.
Even if another instruction is inserted between “0x0000_00FA” and “0x0000_00F5” written to FLCACP, the sector erase or 1-word write is enabled. However, if you write data other than "0x0000_00F5" in FLCACP after writing "0x0000_00FA", it is disabled. Therefore, you must write from "0x0000_00FA" again to enable it. In addition, if you write to FLCACP without executing erase or 1-word write after writing "0x0000_00FA" and "0x0000_00F5", it is disabled regardless of the value. Therefore, you must write "0x0000_00FA" and "0x0000_00F5" in this order again to enable it.

[Note]
If you write ”00” in FLE field with the FLCACP register enabled, it is still maintained as enabled.

16.2.4 Flash-ROM Address Register (FLCADR)

Address: 0x4000_4008
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	FA[16]
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RW
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FA[15:2]														-*	-*
Access	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future extension. "0" is read when reading. Write "0" when writing.

[Description of Register]
FLCADR is a special function register (SFR) to set Flash-ROM rewrite addresses.

[Description of Bits]

- **FA[16:2]** (bit 16-2)
  FA[16:2] is a bit used to set the address for sector erase and 1-word write. When you write the byte address of the flash memory area in this register, the address of the Flash-ROM is set in FA[16:2].

[Note]
You cannot rewrite this register while BUSY bit of FLCSTA register is “1”. In addition, it is prohibition that you use sector erase or 1-word write by setting this register to “0x0001_FE**” (last sector address).

16.2.5 Flash-ROM Write Data Register (FLCWDA)

Address: 0x4000_040C
Access: W
Access size: 32 Bits
Initial Value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	FD[31:16]															
Access	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FD[15:0]															
Access	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
The read data of this register is "0".

[Description of Register]
FLCWDA is a special function register (SFR) to set Flash-ROM rewrite data.

[Description of Bits]

- **FD[31:0]** (bit 31-0)
  FD[31:0] is a bit used to set write data for 1-word write.
  Write to FD[31:0] starts the 1-word write.

[Note]
Clear the contents of the target write addresses in advance. The content of an overwritten address is not guaranteed.
You cannot rewrite this register while BUSY bit of FLCSTA register is “1”. After high-speed clock is selected as the system clock, you can rewrite Flash-ROM data. Rewriting Flash-ROM data while low-speed clock is selected as the system clock cannot be guaranteed. For details of clock selection, see Chapter 6, “Clock”.

16.2.6 Flash-ROM Erase Register (FLCERA)

Address: 0x4000_0410
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	FLE[1:0]	
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future extension. "0" is read when reading. Write "0" when writing.

[Description of Register]
FLCERA is a special function register (SFR) used to start sector erase of Flash-ROM.

[Description of Bits]

- **FLE[1:0]** (bit 1, 0)
  FLE is a bit used to specify the type and start of erase.
  Write to FLE starts erase according to the data. It automatically changes to “00” when the erase is completed. Write to ”00” is prohibited.

FLE[1:0]	Description
00	Erase completed (initial value)
01	Setting prohibited
10	Setting prohibited
11	Start sector erase

[Note]
After high-speed clock is selected as the system clock, you can erase Flash-ROM data. Erasing Flash-ROM sector while low-speed clock is selected as the system clock cannot be guaranteed. For details of clock selection, see Chapter 6, “Clock”.

16.2.7 Flash-ROM Size Register (FLCRSIZ)

Address: 0x4000_0420
Access: R
Access size: 32 Bits
Initial Value: 0x0002_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	FSI[31:16]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSI[15:0]															
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Description of Register]
FLCRSIZ is a special function register (SFR) used to indicate the size of Flash-ROM (in bytes).

[Description of Bits]

- **FSI[31:0]** (bit 31-0)
  FSI[31:0] is a bit used to indicate the size of Flash-ROM (in bytes).

16.2.8 Boot Program Address Register (FLCBADR)

Address: 0x4000_0424
Access: R
Access size: 32 Bits
Initial Value: 0x0001_F000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	BPA[27:16]											
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	BPA[15:12]				-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCBADR is a special function register (SFR) used to indicate the start address of the boot program area.

[Description of Bits]

- **BPA[27:12]** (bit 27-12)
  BPA[27:12] is a bit used to indicate the start address of the boot program area by the initial value.

16.2.9 Set Count Acceptor Register (FLCSCACP)

Address: 0x4000_0460
Access: W
Access size: 32 Bits
Initial Value: 0x0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*	-*
Access	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCAC[7:0]							
Access	-	-	-	-	-	-	-	-	W	W	W	W	W	W	W	W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Note]
*: Reserved bit for future extension. Write "0" when writing.

[Description of Register]
FLCSCACP is a special function register (SFR) used to control (disables/enables) write to the set count registers other than FLCSCACP.

[Description of Bits]

- **FSCAC[7:0]** (bit 7-0)

FSCAC07 to FSCAC00 are registers used to restrict write to set count registers.
FLCSCACP has a state of permitted, requested, or prohibited.

State	Description
Permitted	It is possible to write to Flash-ROM set count registers.
Request	It is impossible to write to Flash-ROM set count registers.
Prohibited	It is impossible to write to Flash-ROM set count registers.

The default is the state of prohibited. The conditions of transition to these states are as follows:

State	Condition of transition
Permitted	Write "0x0000_00FC" data in the requested state.
Request	Write "0x0000_00F3" data in the prohibited state.
Prohibited	Write any data in the permitted state. Write data other than "0x0000_00FC" in the requested data.

16.2.10 Set Count Sector Erase CE Enable Register (FLCSCSCE)

Address: 0x4000_046C
Access: R/W
Access size: 32 Bits
Initial Value: 0x0002_1474

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCSCE[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCSCE[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	1	0	1	0	0	0	1	1	1	0	1	0	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCSCE is a special function register (SFR) used to set the initial value of the erase program counter at the time of selector erase of Flash-ROM.

[Description of Bits]

- **FSCSCE[23:0]** (bit 23-0)
  FSCBCE[23:0] is a bit used to set the initial value of the erase program counter at the time of selector erase of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.11 Set Count Program CE Enable Register (FLCSCPCE)

Address: 0x4000_0470
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_04FB

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPCE[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPCE[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	1	0	0	1	1	1	1	1	0	1	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPCE is a special function register (SFR) used to set the initial value of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPCE[23:0]** (bit 23-0)
  FSCSCE[23:0] is a bit used to set the initial value of the erase program counter at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.12 Set Count Sector Erase WE Enable Register (FLCSCSWE)

Address: 0x4000_047C
Access: R/W
Access size: 32 Bits
Initial Value: 0x0002_13AA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCSWE[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCSWE[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	1	0	0	1	1	1	0	1	0	1	0	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCSWE is a special function register (SFR) used to set the value to start enabling the write signal of the erase program counter at the time of selector erase of Flash-ROM.

[Description of Bits]

- **FSCSWE[23:0]** (bit 23-0)
  For FSCSWE[23:0], the write signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of selector erase of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.13 Set Count Program WE Enable Register (FLCSCPWE)

Address: 0x4000_0480
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0431

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPWE[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPWE[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPWE is a special function register (SFR) used to set the value to start enabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCBWE[23:0]** (bit 23-0)
  For FSCPWE[23:0], the write signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.14 Set Count Program PROG2_1st Enable Register (FLCSCPP21E)

Address: 0x4000_0484
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_03DD

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP21E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP21E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	1	1	1	0	1	1	1	0	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP21E is a special function register (SFR) used to set the value to start enabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP21E[23:0]** (bit 23-0)
  For FSCPP21E[23:0], the write signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.15 Set Count Program PROG2_1st Disable Register (FLCSCPP21D)

Address: 0x4000_0488
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0313

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP21D[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP21D[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	1	0	0	0	1	0	0	1	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP21D is a special function register (SFR) used to set the value to start disabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP21D[23:0]** (bit 23-0)
  For FSCPP21D23 to FSCPP21D00, the write signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.16 Set Count Program PROG2_2nd Enable Register (FLCSCPP22E)

Address: 0x4000_048C
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0311

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP22E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP22E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP22E is a special function register (SFR) used to set the value to start enabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP22E[23:0]** (bit 23-0)
  For FSCPP22E[23:0], the write signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.17 Set Count Program PROG2_2nd Disable Register (FLCSCPP22D)

Address: 0x4000_0490
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0247

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP22D[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP22D[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP22D is a special function register (SFR) used to set the value to start disabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP22D[23:0]** (bit 23-0)
  For FSCPP22D[23:0], the write signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.18 Set Count Program PROG2_3rd Enable Register (FLCSCPP23E)

Address: 0x4000_0494
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0245

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP23E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP23E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP23E is a special function register (SFR) used to set the value to start enabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP23E[23:0]** (bit 23-0)
  For FSCPP23E[23:0], the write signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.19 Set Count Program PROG2_3rd Disable Register (FLCSCPP23D)

Address: 0x4000_0498
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_017B

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP22D[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP22D[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	1	0	1	1	1	1	0	1	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP23D is a special function register (SFR) used to set the value to start disabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP23D[23:0]** (bit 23-0)
  For FSCPP23D[23:0], the write signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.20 Set Count Program PROG2_4th Enable Register (FLCSCPP24E)

Address: 0x4000_049C
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0179

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP24E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP24E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	1	0	1	1	1	1	0	0	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP24E is a special function register (SFR) used to set the value to start enabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP24E[23:0]** (bit 23-0)
  For FSCPP24E[23:0], the write signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.21 Set Count Program PROG2_4th Disable Register (FLCSCPP24D)

Address: 0x4000_04A0
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_00AF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPP24D[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPP24D[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	0	1	0	1	1	1	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPP24D is a special function register (SFR) used to set the value to start disabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPP24D[23:0]** (bit 23-0)
  For FSCPP24D[23:0], the write signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.22 Set Count Program BYTE1 Enable Register (FLCSCPB1E)

Address: 0x4000_04A8
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0312

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCSCPB1E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCSCPB1E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	1	0	0	0	1	0	0	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPB1E is a special function register (SFR) used to set the value to start enabling bit 1 of byte signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- FSCSCPB1E[23:0] (bit 23-0)
  For FSCSCPB1E[23:0], bit 1 of the byte signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM. In addition, the other bits of the byte signal are disabled.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.23 Set Count Program BYTE2 Enable Register (FLCSCPB2E)

Address: 0x4000_04AC
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0246

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCSCPB2E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCSCPB2E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPB2E is a special function register (SFR) used to set the value to start enabling bit 2 of byte signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCSCPB2E[23:0]** (bit 23-0)
  For FSCSCPB2E[23:0], bit 2 of the byte signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM. In addition, the other bits of the byte signal are disabled.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.24 Set Count Program BYTE3 Enable Register (FLCSCPB3E)

Address: 0x4000_04B0
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_017A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCSCPB3E[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCSCPB3E[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	1	0	1	1	1	1	0	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPB3E is a special function register (SFR) used to set the value to start enabling bit 3 of byte signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCSCPB3E[23:0]** (bit 23-0)
  For FSCSCPB3E[23:0], bit 3 of the byte signal to Flash-ROM is enabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM. In addition, the other bits of the byte signal are disabled.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.25 Set Count Erase WE Disable Register (FLCSCEWED)

Address: 0x4000_04B4
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_06AA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCEWED[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCEWED[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	1	1	0	1	0	1	0	1	0	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCEWED is a special function register (SFR) used to set the value to start disabling the write signal of the erase program counter at the time of erase of Flash-ROM.

[Description of Bits]

- **FSCEWED[23:0]** (bit 23-0)
  For FSCEWED[23:0], the write signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of erase of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.26 Set Count Program WE Disable Register (FLCSCPWED)

Address: 0x4000_04B8
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_00AE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCPWED[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCPWED[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	0	1	0	1	1	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCPWED is a special function register (SFR) used to set the value to start disabling the write signal of the erase program counter at the time of programming of Flash-ROM.

[Description of Bits]

- **FSCPWED[23:0]** (bit 23-0)
  For FSCPWED[23:0], the write signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.27 Set Count CE Disable Register (FLCSCCED)

Address: 0x4000_04BC
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCCED[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCCED[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCCED is a special function register (SFR) used to set the value to start disabling the chip enable signal of the erase program counter at the time of erase or programming of Flash-ROM.

[Description of Bits]

- **FSCCCED[23:0]** (bit 23-0)
  For FSCCED[23:0], the chip enable signal to Flash-ROM is disabled when the erase program counter value matches with the bit value at the time of erase or programming of Flash-ROM.
  For the setting value, see 16.3.2 “Counter Setting”.

16.2.28 Set Count Termination Register (FLCSCEND)

Address: 0x4000_04C0
Access: R/W
Access size: 32 Bits
Initial Value: 0x0000_0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol name	-*	-*	-*	-*	-*	-*	-*	-*	FSCCEND[23:16]							
Access	-	-	-	-	-	-	-	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol name	FSCCEND[15:0]															
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

[Note]
*: Reserved bit for future extension. "0" is read when reading.

[Description of Register]
FLCSCEND is a special function register (SFR) used to set the completion judgment value of the erase program counter at the time of erase or programming of Flash-ROM.

[Description of Bits]

- **FSCCEND[23:0]** (bit 23-0)

FSCCEND[23:0] terminates the erase or programming processing when the erase program counter value matches with the bit value at the time of erase or programming of Flash-ROM.
For the setting value, see 16.3.2 “Counter Setting”.

16.3 Description of Operation

16.3.1 Erase/Write Flash-ROM

It is possible to execute sector erase and 1-word write. It is needed to access the register of the Flash-ROM controller from the CPU according to the procedure.

It includes the flash rewrite acceptor function which restricts the rewrite operation to prevent an improper rewrite to Flash-ROM. Writing of "0x0000_00FA" and "0x0000_00F5" to the Flash ROM acceptor register (FLCACP) in this order enables sector erase or 1-word write only once.

[Note]
Software reset during Flash-ROM erase/write is prohibited. A program that executes Flash-ROM erase/write should not be placed on Flash-ROM.
The last sector is the test code area. Erase/Program operation is prohibited.

16.3.2 Counter Setting

Set the timing of the control signal to Flash-ROM in the set count register according to the clock cycle when executing sector erase or 1-word write. The initial value supports the clock cycle of 32MHz. The setting values of the set count registers are shown in Table 16-1.

Table 16-1 Setting Values of Set Count Registers for Clock Cycles

Register name	Setting value of the set count register by clock cycle
FLCSCSCE	0x0002_1474
FLCSCPCE	0x0000_04FB
FLCSCSWE	0x0002_13AA
FLCSCPWE	0x0000_0431
FLCSCPP21E	0x0000_03DD
FLCSCPP21D	0x0000_0313
FLCSCPP22E	0x0000_0311
FLCSCPP22D	0x0000_0247
FLCSCPP23E	0x0000_0245
FLCSCPP23D	0x0000_017B
FLCSCPP24E	0x0000_0179
FLCSCPP24D	0x0000_00AF
FLCSCPB1E	0x0000_0312
FLCSCPB2E	0x0000_0246
FLCSCPB3E	0x0000_017A
FLCSCEWED	0x0000_06AA
FLCSCPWED	0x0000_00AE
FLCSCCED	0x0000_0006
FLCSCEND	0x0000_0001

16.3.3 Sector Erase

This function erases data in the main area of Flash-ROM by sector.

Erase of the specified sector data is started when you write “0x0000_00FA” and “0x0000_00F5” to Flash ROM acceptor register (FLCACP), set the sector address to the Flash ROM address register (FLCADR), and write “11” to FLE bit of Flash ROM erase register (FLCERA). The FLCSTA register is “0x0000_0001” during erase. When erase is completed, the FLCSTA register changes to “0x0000_000”.

Erase/write processing should be executed by a program that is loaded on a memory other than the target Flash-ROM.It prevents the rewriting program from being lost if it is on Flash-ROM, due to an unintended operation during erase/write processing.

The CPU enters the waiting state when reading of Flash-ROM occurs during erase. Write access to the register during erase is prohibited.

Figure 16-1 shows the processing flow of block erase in the main area. Read the selected block or all data of the sector for check after the erase is completed.

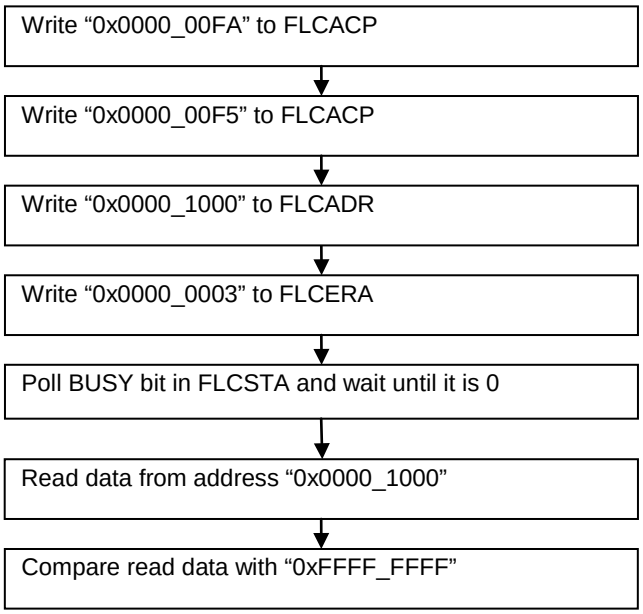

Write “0x0000_00FA” to FLCACP
Write “0x0000_00F5” to FLCACP
Write “0x0000_1000” to FLCADR
Write “0x0000_0003” to FLCERA
Poll BUSY bit in FLCSTA and wait until it is 0
Read data from address “0x0000_1000”
Compare read data with “0xFFFF_FFFF”

Figure 16-1 Program Flow of Sector Erase (When Executed on RAM)

16.3.4 1-word Write

This function writes data in the main area of Flash-ROM in 4 bytes.

Write to the specified address is started when you write "0x0000_00FA" and "0x0000_00F5" to Flash ROM acceptor register (FLCACP), set the address to the Flash ROM address register (FLCADR), and write data to Flash ROM write data register (FLCWDA). During 1-word write, the FLCSTA register is “0x0000_0001”. When write is completed, the FLCSTA register changes to “0x0000_0000”.

Erase/write processing should be executed by a program that is loaded on a memory other than the target Flash-ROM.

The CPU enters the waiting state when reading of Flash-ROM occurs during write. Write access to the register during write is prohibited.

Figure 16-2 shows the program flow of 1-word write when executing the code on Flash-ROM. Check data when the writing is completed.

[Note]
Data should be written to an erased area on Flash-ROM. In addition, when you want to rewrite data that is written once, erase it before writing it again.

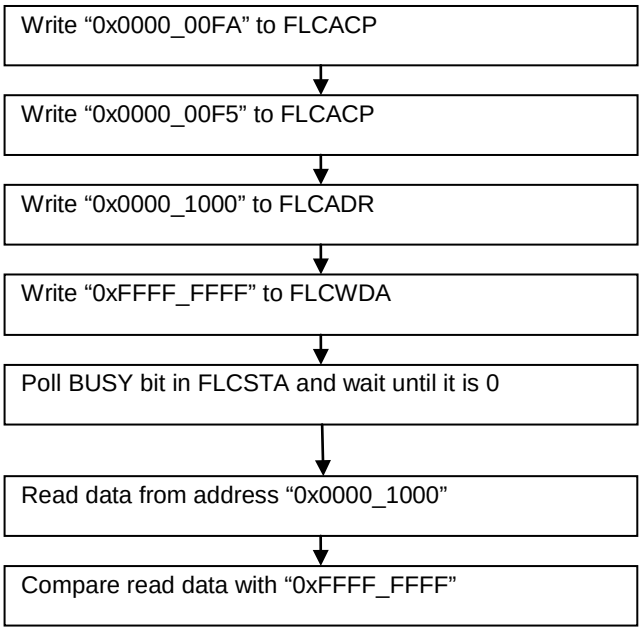

Write “0x0000_00FA” to FLCACP
Write “0x0000_00F5” to FLCACP
Write “0x0000_1000” to FLCADR
Write “0xFFFF_FFFF” to FLCWDA
Poll BUSY bit in FLCSTA and wait until it is 0
Read data from address “0x0000_1000”
Compare read data with “0xFFFF_FFFF”

Figure 16-2 Flow of 1-word Write (When Execute on RAM)

16.3.5 Erase/Write to Area Where Flash-ROM Is Not Implemented

It is prohibited to specify an area where Flash-ROM is not implemented to execute sector erase or 1-word write. If it is executed, the Flash-ROM is not updated. The BUSY field of the Flash ROM status register (FLCSTA) changes to ”1”, but goes back to ”0” in a short time because the Flash-ROM is not updated.

16.3.6 Notes in Use

If the power is down or the operation is terminated forcibly during sector erase or 1-word write, retry the sector erase and rewrite the area.

Chapter 17 On-Chip Debug Function

17. On-Chip Debug Function

17.1 Overview

This LSI implements a SW-DP (serial wire debug port) as the debug interface.
The connection example is shown in Figure17-1. For details, see the debugger manual.

[Note] The timer and WDT clock are stopped at break in the debugger.

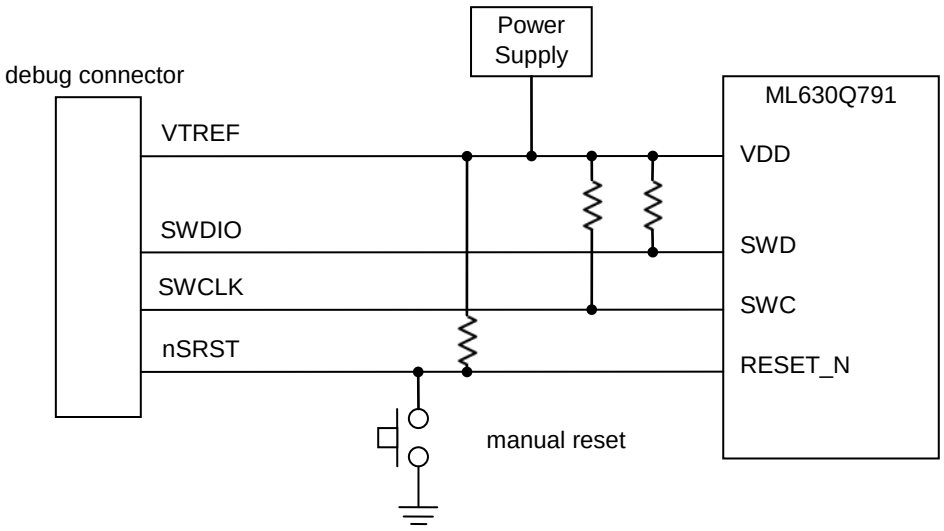


Figure17 -1 Connection with Debug Connector

Chapter 18 Power Supply Circuit

18. Power Supply Circuit

18.1 Overview

This LSI includes a voltage regulator for the internal logic.

18.1.1 Features

GPIO and the regulator operate with voltage supplied from VDD. Internal logic circuit, FLASH, RAM, FLL operate with voltage that the regulator outputs to VDDL.

18.1.2 Configuration

Figure 18-1 shows the configuration of the power supply circuit.

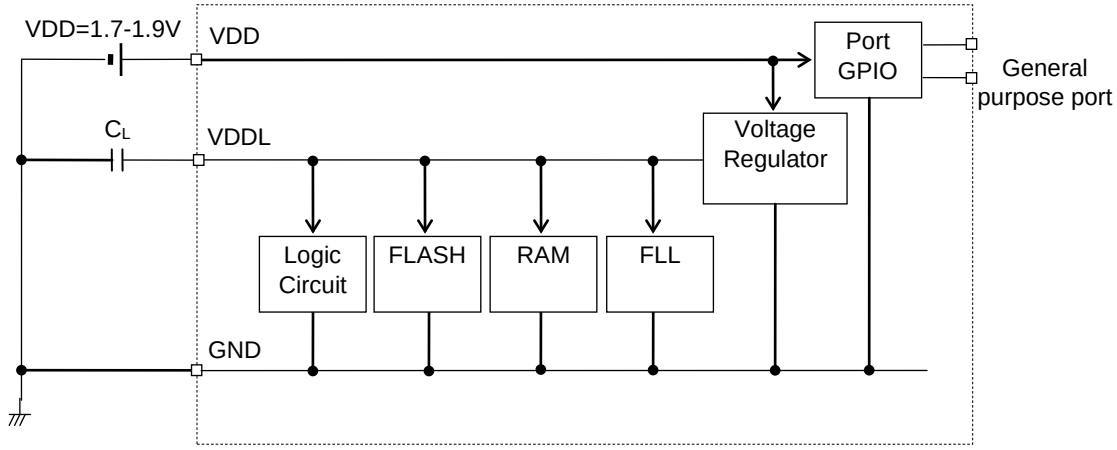


Figure 18-1 Configuration of Power Supply Circuit

18.1.3 List of Pins

Pin name	I/O	Description
VDDL	-	Positive power supply pin for the internal logic circuits

Appendixes

Appendix A Registers

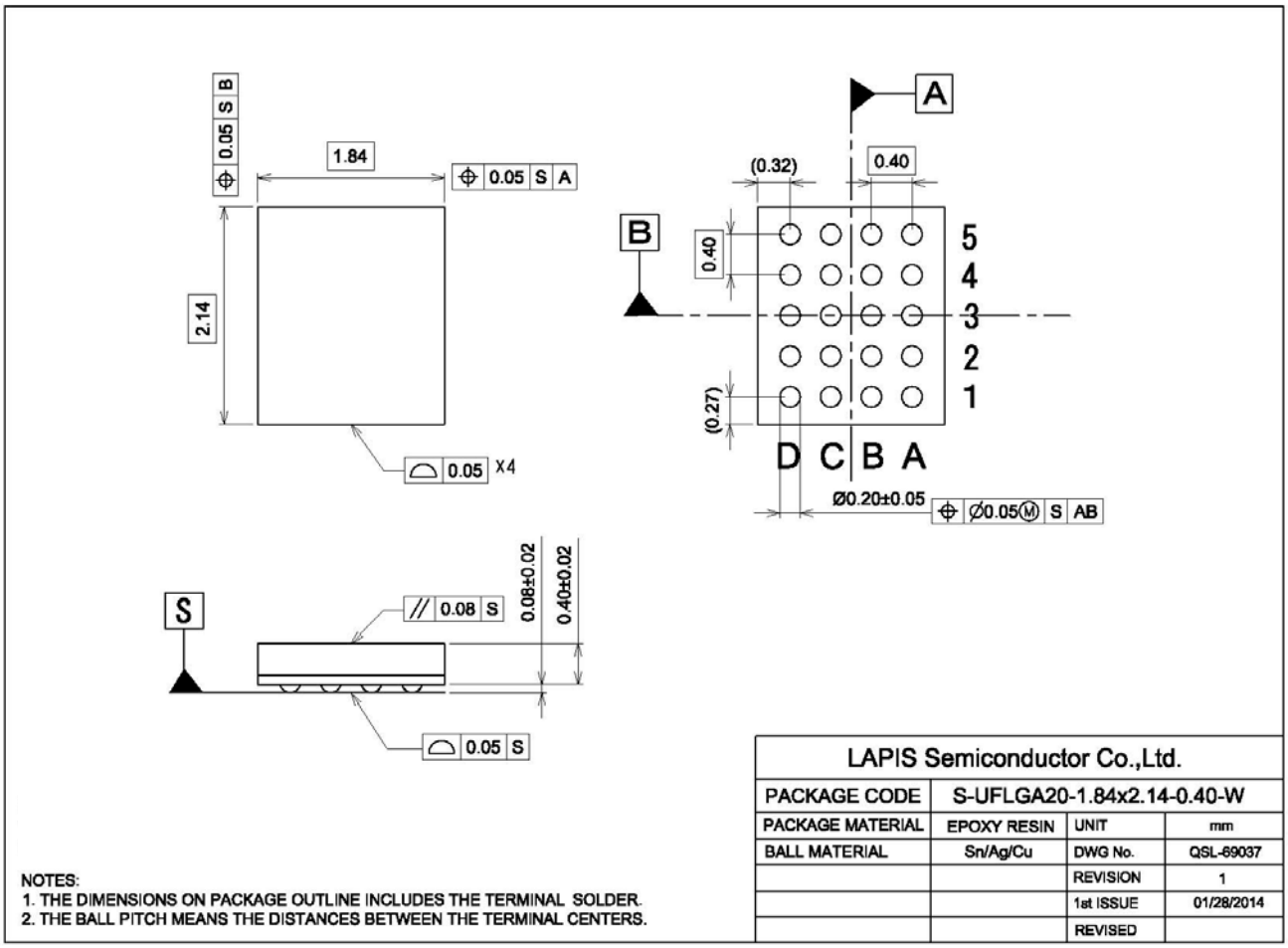
Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4000_0010	Remapping control register	SYSCON_REMAP_CON	R/W	32	0x0000_0000
0x4000_0014	Remapping base address register	SYSCON_REMAP_BASE	R/W	32	0x1001_F000
0x4000_0200	Revision register	IDR	R	32	0x0630_7900
0x4000_0220	Peripheral clock enable register	PECLKEN	R/W	32	0x8000_0000
0x4000_0224	Peripheral clock disable register	PECLKDIS	R/W	32	0x0113_11FF
0x4000_0228	Peripheral reset enable register	PERSTEN	R/W	32	0x0113_11FF
0x4000_022C	Peripheral reset disable register	PERSTDIS	R/W	32	0x8000_0000
0x4000_0260	Port A mode setting register	PAMOD	R/W	32	0x0000_0000
0x4000_0300	Frequency control register 0	FCON0	R/W	32	0x0000_0000
0x4000_0304	Frequency control register 1	FCON1	R/W	32	0x0000_0002
0x4000_0400	Flash-ROM status register	FLCSTA	R	32	0x0000_0000
0x4000_0404	Flash-ROM acceptor register	FLCACP	W	32	0x0000_0000
0x4000_0408	Flash-ROM address register	FLCADR	R/W	32	0x0000_0000
0x4000_040C	Flash-ROM write data register	FLCWDA	W	32	0x0000_0000
0x4000_0410	Flash-ROM erase register	FLCERA	R/W	32	0x0000_0000
0x4000_0420	Flash-ROM size register	FLCRSIZ	R	32	0x0002_0000
0x4000_0424	Boot program address register	FLCBADR	R	32	0x0001_F000
0x4000_0460	Set count acceptor register	FLCSCACP	W	32	0x0000_0000
0x4000_046C	Set count sector erase CE enable register	FLCSCSCE	R/W	32	0x0002_1474
0x4000_0470	Set count program CE enable register	FLCSCPCE	R/W	32	0x0000_04FB
0x4000_047C	Set count sector erase WE enable register	FLCSCSWE	R/W	32	0x0002_13AA
0x4000_0480	Set count program WE enable register	FLCSCPWE	R/W	32	0x0000_0431
0x4000_0484	Set count program PROG2_1st enable register	FLCSCPP21E	R/W	32	0x0000_03DD
0x4000_0488	Set count program PROG2_1st disable register	FLCSCPP21D	R/W	32	0x0000_0313
0x4000_048C	Set count program PROG2_2nd enable register	FLCSCPP22E	R/W	32	0x0000_0311
0x4000_0490	Set count program PROG2_2nd disable register	FLCSCPP22D	R/W	32	0x0000_0247
0x4000_0494	Set count program PROG2_3rd enable register	FLCSCPP23E	R/W	32	0x0000_0245
0x4000_0498	Set count program PROG2_3rd disable register	FLCSCPP23D	R/W	32	0x0000_017B
0x4000_049C	Set count program PROG2_4th enable register	FLCSCPP24E	R/W	32	0x0000_0179
0x4000_04A0	Set count program PROG2_4th disable register	FLCSCPP24D	R/W	32	0x0000_00AF
0x4000_04A8	Set count program BYTE1 enable register	FLCSCPB1E	R/W	32	0x0000_0312
0x4000_04AC	Set count program BYTE2 enable register	FLCSCPB2E	R/W	32	0x0000_0246
0x4000_04B0	Set count program BYTE3 enable register	FLCSCPB3E	R/W	32	0x0000_017A
0x4000_04B4	Set count erase WE disable register	FLCSCEWED	R/W	32	0x0000_06AA
0x4000_04B8	Set count program WE disable register	FLCSCPWED	R/W	32	0x0000_00AE
0x4000_04BC	Set count CE disable register	FLCSCCED	R/W	32	0x0000_0006
0x4000_04C0	Set count termination register	FLCSCEND	R/W	32	0x0000_0001
0x4000_1400	Timer 0 data register	TM0D	R/W	32	0x0000_00FF
0x4000_1404	Timer 0 counter register	TM0C	R/W	32	0x0000_0000
0x4000_1408	Timer 0 control register 0	TM0CON0	R/W	32	0x0000_0000
0x4000_140C	Timer 0 control register 1	TM0CON1	R/W	32	0x0000_0000
0x4000_1410	Timer 1 data register	TM1D	R/W	32	0x0000_00FF
0x4000_1414	Timer 1 counter register	TM1C	R/W	32	0x0000_0000

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4000_1418	Timer 1 control register 0	TM1CON0	R/W	32	0x0000_0000
0x4000_141C	Timer 1 control register 1	TM1CON1	R/W	32	0x0000_0000
0x4000_1800	Timer 2 data register	TM2D	R/W	32	0x0000_00FF
0x4000_1804	Timer 2 counter register	TM2C	R/W	32	0x0000_0000
0x4000_1808	Timer 2 control register 0	TM2CON0	R/W	32	0x0000_0000
0x4000_180C	Timer 2 control register 1	TM2CON1	R/W	32	0x0000_0000
0x4000_1810	Timer 3 data register	TM3D	R/W	32	0x0000_00FF
0x4000_1814	Timer 3 counter register	TM3C	R/W	32	0x0000_0000
0x4000_1818	Timer 3 control register 0	TM3CON0	R/W	32	0x0000_0000
0x4000_181C	Timer 3 control register 1	TM3CON1	R/W	32	0x0000_0000
0x4000_1C00	Timer 4 data register	TM4D	R/W	32	0x0000_00FF
0x4000_1C04	Timer 4 counter register	TM4C	R/W	32	0x0000_0000
0x4000_1C08	Timer 4 control register 0	TM4CON0	R/W	32	0x0000_0000
0x4000_1C0C	Timer 4 control register 1	TM4CON1	R/W	32	0x0000_0000
0x4000_1C10	Timer 5 data register	TM5D	R/W	32	0x0000_00FF
0x4000_1C14	Timer 5 counter register	TM5C	R/W	32	0x0000_0000
0x4000_1C18	Timer 5 control register 0	TM5CON0	R/W	32	0x0000_0000
0x4000_1C1C	Timer 5 control register 1	TM5CON1	R/W	32	0x0000_0000
0x4000_2000	Timer 6 data register	TM6D	R/W	32	0x0000_00FF
0x4000_2004	Timer 6 counter register	TM6C	R/W	32	0x0000_0000
0x4000_2008	Timer 6 control register 0	TM6CON0	R/W	32	0x0000_0000
0x4000_200C	Timer 6 control register 1	TM6CON1	R/W	32	0x0000_0000
0x4000_2010	Timer 7 data register	TM7D	R/W	32	0x0000_00FF
0x4000_2014	Timer 7 counter register	TM7C	R/W	32	0x0000_0000
0x4000_2018	Timer 7 control register 0	TM7CON0	R/W	32	0x0000_0000
0x4000_201C	Timer 7 control register 1	TM7CON1	R/W	32	0x0000_0000
0x4000_A000	GPIO data register	PIODAT	R/W	32	0x0000_0000
0x4000_A004	GPIO direction register	PIODIR	R/W	32	0x0000_0000
0x4000_A008	GPIO port control register	PIOCON	R/W	32	0x0000_0000
0x4000_A010	GPIO interrupt enable register	PIOIE	R/W	32	0x0000_0000
0x4000_A014	GPIO interrupt mode register	PIOIM	R/W	32	0x0000_0000
0x4000_A018	GPIO interrupt status register	PIOIS	R/W	32	0x0000_0000
0x4001_0400	Watchdog timer control register	WDTCON	R/W	32	0x0000_0000
0x4001_0404	Watchdog timer mode register	WDTMOD	R/W	32	0x0000_0002
0x4001_4400	PWM0 cycle register	PW0P	R/W	32	0x0000_FFFF
0x4001_4404	PWM0 duty register	PW0D	R/W	32	0x0000_0000
0x4001_4408	PWM0 counter register	PW0C	R/W	32	0x0000_0000
0x4001_440C	PWM0 control register 0	PW0CON0	R/W	32	0x0000_0000
0x4001_4410	PWM0 control register 1	PW0CON1	R/W	32	0x0000_0040
0x4004_0000	High-speed time base counter frequency divide register	HTBDR	R/W	32	0x0000_0000
0x4004_5000	Operation status register	CALSTS	R/W	32	0x0000_0000
0x4004_5008	Operation input register AL	CALAL	R/W	32	0x0000_0000
0x4004_500C	Operation input register AH	CALAH	R/W	32	0x0000_0000
0x4004_5010	Operation input register BL	CALBL	R/W	32	0x0000_0000
0x4004_5014	Operation input register BH	CALBH	R/W	32	0x0000_0000
0x4004_5018	Calculation result register 0L	CALR0L	R	32	0x0000_0000

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4004_501C	Calculation result register 0H	CALR0H	R	32	0x0000_0000
0x4004_5020	Calculation result register 1L	CALR1L	R	32	0x0000_0000
0x4004_5024	Calculation result register 1H	CALR1H	R	32	0x0000_0000
0x4005_0000	Configuration register	HIFCFG	R/W	32	0xFFFF_2F00
0x4005_0008	Operation status register	HIFST	R/W	32	0x0000_FE00
0x4005_000C	Interrupt request register	HIFRQ	R/W	32	0x0000_0000
0x4005_0010	FIFO register	HIFFIFOW	R/W	32/8	0x0000_00XX
0x4005_0014	FIFO switch register	HIFFSEL	R/W	32	0x0000_0000
0x4005_0018	FIFO write pointer register	HIFWP	R/W	32	0x0000_0000
0x4005_001C	FIFO read pointer register	HIFRP	R/W	32	0x0000_0000
0x4005_0020	Parameter register F	HIFPRMF	R	32	0x0000_0000
0x4005_0024	Parameter register B	HIFPRMB	R	32	0x0000_0000
0x4005_0028	Parameter register 7	HIFPRM7	R	32	0x0000_0000
0x4005_002C	Parameter register 3	HIFPRM3	R	32	0x0000_0000
0x4005_0030	Command register	HIFCMD	R	32	0x0000_0000
0x4005_0040	Result register 00	HIFRLT00W	R/W	32/16/8	0x0000_0000
0x4005_0044	Result register 04	HIFRLT04W	R/W	32/16/8	0x0000_0000
0x4005_0048	Result register 08	HIFRLT08W	R/W	32/16/8	0x0000_0000
0x4005_004C	Result register 0C	HIFRLT0CW	R/W	32/16/8	0x0000_0000
0x4005_0050	Result register 10	HIFRLT10W	R/W	32/16/8	0x0000_0000
0x4005_0054	Result register 14	HIFRLT14W	R/W	32/16/8	0x0000_0000
0x4005_0058	Result register 18	HIFRLT18W	R/W	32/16/8	0x0000_0000
0x4005_005C	Result register 1C	HIFRLT1CW	R/W	32/16/8	0x0000_0000
0x4005_0060	Result register 20	HIFRLT20W	R/W	32/16/8	0x0000_0000
0x4005_0064	Result register 24	HIFRLT24W	R/W	32/16/8	0x0000_0000
0x4005_0068	Result register 28	HIFRLT28W	R/W	32/16/8	0x0000_0000
0x4005_006C	Result register 2C	HIFRLT2CW	R/W	32/16/8	0x0000_0000
0x4005_0070	Result register 30	HIFRLT30W	R/W	32/16/8	0x0000_0000
0x4005_0074	Result register 34	HIFRLT34W	R/W	32/16/8	0x0000_0000
0x4005_0078	Result register 38	HIFRLT38W	R/W	32/16/8	0x0000_0000
0x4005_007C	Result register 3C	HIFRLT3CW	R/W	32/16/8	0x0000_0000
0x4008_1000	UART receive data register	UARTRBR	R	32	Undefined
	UART transmit data register	UARTTHR	W		Undefined
	UART baud rate dividing register (LSB)	UARTDLL	R/W		0x0000_0000
0x4008_1004	UART interrupt enable register	UARTIER	R/W	32	0x0000_0000
	UART baud rate dividing register (MSB)	UARTDLM	R/W		0x0000_0000
0x4008_1008	UART interrupt status register	UARTIIR	R	32	0x0000_0001
	UART FIFO control register	UARTFCR	W		0x0000_0000
0x4008_100C	UART line control register	UARTLCR	R/W	32	0x0000_0000
0x4008_1014	UART line status register	UARTLSR	R	32	0x0000_0060
0x4008_101C	UART scratchpad register	UARTSCR	R/W	32	0x0000_0000
0x4008_3004	I^2C control register 0	I2CCTL0	R/W	32	0x0000_0000
0x4008_3008	I^2C status register 0	I2CSR0	R/W	32	0x0000_0000
0x4008_300C	I^2C data register 0	I2CDR0	R/W	32	0x0000_0000
0x4008_3010	I^2C bus monitor register 0	I2CMON0	R	32	0x0000_0003
0x4008_3014	I^2C bus transfer rate setup counter 0	I2CBC0	R/W	32	0x0000_0000
0x4008_3018	I^2C mode register 0	I2CMOD0	R/W	32	0x0000_0000

Address	Name	Symbol	R/W	Size [bits]	Initial value
0x4008_301C	I²C buffer mode slave address register 0	I2CBUFSLV0	R/W	32	0x0000_0000
0x4008_3020	I²C buffer mode sub address register 0	I2CBUFSUB0	R/W	32	0x0000_0000
0x4008_3024	I²C buffer mode format register 0	I2CBUFFOR0	R/W	32	0x0000_0000
0x4008_3028	I²C buffer mode control register 0	I2CBUFCTL0	R/W	32	0x0000_0000
0x4008_302C	I²C buffer mode interrupt mask register 0	I2CBUFMSK0	R/W	32	0x0000_0000
0x4008_3030	I²C buffer mode status register 0	I2CBUFSTA0	R/W	32	0x0000_0000
0x4008_3034	I²C buffer mode level register 0	I2CBUFLEV0	R/W	32	0x0000_0000
0x4008_3048	I²C timer register 0	I2CTMR0	R/W	32	0x0000_0000
0x4008_3050	I²C input noise filter setting register 0	I2CNF0	R/W	32	0x0000_0001
0x4008_3404	I²C control register 1	I2CCTL1	R/W	32	0x0000_0000
0x4008_3408	I²C status register 1	I2CSR1	R/W	32	0x0000_0000
0x4008_340C	I²C data register 1	I2CDR1	R/W	32	0x0000_0000
0x4008_3410	I²C bus monitor register 1	I2CMON1	R	32	0x0000_0003
0x4008_3414	I²C bus transfer rate setup counter 1	I2CBC1	R/W	32	0x0000_0000
0x4008_3418	I²C mode register 1	I2CMOD1	R/W	32	0x0000_0000
0x4008_341C	I²C buffer mode slave address register 1	I2CBUFSLV1	R/W	32	0x0000_0000
0x4008_3420	I²C buffer mode sub address register 1	I2CBUFSUB1	R/W	32	0x0000_0000
0x4008_3424	I²C buffer mode format register 1	I2CBUFFOR1	R/W	32	0x0000_0000
0x4008_3428	I²C buffer mode control register 1	I2CBUFCTL1	R/W	32	0x0000_0000
0x4008_342C	I²C buffer mode interrupt mask register 1	I2CBUFMSK1	R/W	32	0x0000_0000
0x4008_3430	I²C buffer mode status register 1	I2CBUFSTA1	R/W	32	0x0000_0000
0x4008_3434	I²C buffer mode level register 1	I2CBUFLEV1	R/W	32	0x0000_0000
0x4008_3448	I²C timer register 1	I2CTMR1	R/W	32	0x0000_0000
0x4008_3450	I²C input noise filter setting register 1	I2CNF1	R/W	32	0x0000_0001
0xE000_E100	Interrupt set-enable register	NVIC_ISER	R/W	32	0x0000_0000
0xE000_E180	Interrupt clear-enable register	NVIC_ICER	R/W	32	0x0000_0000
0xE000_E200	Interrupt set-pending register	NVIC_ISPR	R/W	32	0x0000_0000
0xE000_E280	Interrupt clear-pending register	NVIC_ICPR	R/W	32	0x0000_0000
0xE000_E400	Interrupt priority register 0	NVIC_IPR0	R/W	32	0x0000_0000
0xE000_E404	Interrupt priority register 1	NVIC_IPR1	R/W	32	0x0000_0000
0xE000_E408	Interrupt priority register 2	NVIC_IPR2	R/W	32	0x0000_0000
0xE000_E40C	Interrupt priority register 3	NVIC_IPR3	R/W	32	0x0000_0000
0xE000_E410	Interrupt priority register 4	NVIC_IPR4	R/W	32	0x0000_0000
0xE000_E414	Interrupt priority register 5	NVIC_IPR5	R/W	32	0x0000_0000
0xE000_E418	Interrupt priority register 6	NVIC_IPR6	R/W	32	0x0000_0000
0xE000_E41C	Interrupt priority register 7	NVIC_IPR7	R/W	32	0x0000_0000
0xE000_ED00	CPUID register	CPUID	R	32	0x410C_C200
0xE000_ED04	Interrupt control and state register	ICSR	R/W	32	0x0000_0000
0xE000_ED0C	Application interrupt and reset control register	AIRCR	R/W	32	0xFA05_0000
0xE000_ED10	System control register	SCR	R/W	32	0x0000_0000
0xE000_ED14	Configuration and control register	CCR	R	32	0x0000_0208
0xE000_ED1C	System handler priority register 2	SHPR2	R/W	32	0x0000_0000
0xE000_ED20	System handler priority register 3	SHPR3	R/W	32	0x0000_0000

Appendix B Package Dimensions



Notes for Mounting the Surface Mount Type Package

The surface mount type packages are very susceptible to heat in reflow mounting and humidity absorbed in storage. Therefore, before you perform reflow mounting, contact a ROHM sales office for the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature and times).

Appendix C Electrical Characteristics

●Absolute Maximum Ratings

(GND=0V)

Parameter	Symbol	Condition	Rating	Unit
Power supply voltage (Digital I/O)	V_{DD}	Ta=25°C	-0.3 to 4.6	V
Power supply voltage (Digital CORE)	V_{DDL}	Ta=25°C	-0.3 to 1.8	V
Input voltage	V_{IN}	Ta=25°C	-0.3 to 4.6	V
Output voltage	V_{OUT}	Ta=25°C	-0.3 to 4.6	V
Output current	I_{OUT}	Ta=25°C	-10 to 10	mA
Power dissipation	PD	Ta=25°C	0.8	W
Storage temperature	T_{STG}	—	-55 to 150	°C

●Recommended Operating Conditions

(GND=0V)

Parameter	Symbol	Condition	Range	Unit
Ambient temperature	T_a	—	-40 to 85	°C
Power supply voltage	V_{DD}	—	1.7 to 1.9	V
Input voltage	V_{IN0}	—	0 to V_{DD}	V
	V_{IN1}	*1	0 to 3.6	V
Input clock frequency	f_{CLK}	—	32.768±1%	kHz
VDDL pin external capacitance	C_L	—	2.2±50%	µF

*1 SCL_S, SDA_S, SDA0_M, SCL0_M, PA0, PA1 using as I^2C bus interface.

●Operating Conditions of Flash Memory

(GND=0V)

Parameter	Symbol	Condition	Range	Unit
Ambient temperature	Ta	—	-40 to 85	°C
Power supply voltage	V_{DD}	—	1.7 to 1.9	V
Rewrite count	C_{EP}	—	1000	times
Data retention	Y_{DR}	—	10	years

●DC Characteristics (1/2)

(V_{DD}=1.7 to 1.9V, GND=0V, Ta=-40 to 85°C)

Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
Power consumption (Sleep)	IDD2	CPU stop *1	—	2.5	120	μA
Power consumption (Low-speed operation)	IDD3	CPU 32.768kHz operation *1	—	0.5	0.7	mA
Power consumption (High-speed operation)	IDD4	CPU 32MHz operation	—	5.0	6.5	mA
Power consumption (At reset)	IDD-R	RESETN pin is Low	—	0.4	0.6	mA

*1 operate with the low-speed clock and stop the high-speed clock (FLL). Peripherals except HostIF are initial state.

●DC Characteristics (2/2)

(V_{DD}=1.7 to 1.9V, GND=0V, Ta=-40 to 85°C)

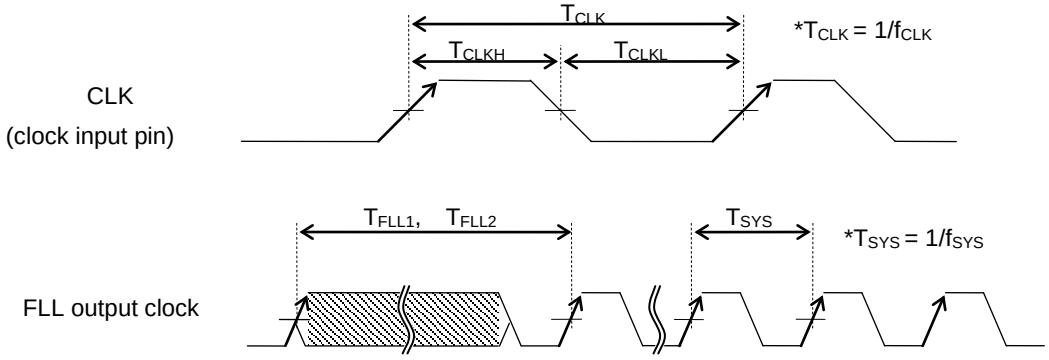
Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
Output voltage 1 (SDA0_M, SCL0_M PA0*1, PA1*1 SCL_S*2, SDA_S*2)	VOH1	—	—	—	—	V
	VOL1	IOL = 3mA	—	—	V_{DD} x 0.2	
Output voltage 2	VOH2	IOH = -2mA	V_{DD} - 0.45	—	—	V
(Other pins)	VOL2	IOL = 2mA	—	—	0.45	
Output leakage	IOOH	VOH = V_{DD} (in high-impedance state)	—	—	1	μA
	IOOL	VOL = 0V (in high-impedance state)	-1	—	—	
input current 1	IIH1Z	VIH = V_{DD}	—	—	1	μA
	IIL1Z	VIL = GND	-1	—	—	
	IIH1	VIH = V_{DD} (pull-down)	2	—	200	
	IIL1	VIL = GND (pull-up)	-200	—	-2	
Input voltage	VIH1	—	V_{DD} x 0.7	—	—	V
	VIL1	—	—	—	V_{DD} x 0.3	

*1 Output voltage 1 shows the characteristic in case the secondary pin function (I2C) is selected.
*2 Output voltage 1 shows the characteristic in case I2C is selected for host interface.

●AC Characteristics (Clock)

(V_{DD}=1.7 to 1.9V,GND= 0V, Ta=-40 to 85°C)

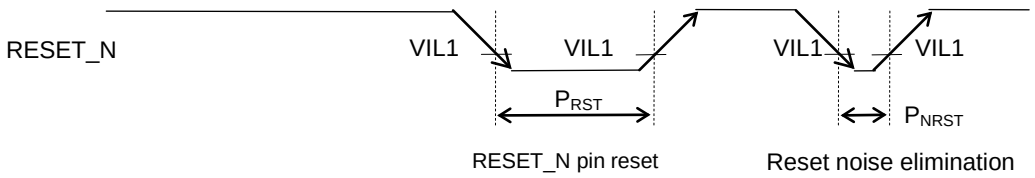
Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
Input clock frequency	f_{CLK}	—	Typ. -1%	32.768	Typ. +1%	kHz
Input clock High pulse width	T_{CLKH}	—	Typ. -1%	15.259	Typ. +1%	µs
Input clock Low pulse width	T_{CLKL}	—	Typ. -1%	15.259	Typ. +1%	µs
System clock frequency	f_{SYS}	f_{CLK} = 32.768kHz	Typ -5%	32	Typ +5%	MHz
FLL activation time (Normal activation)	T_{FLL1}	f_{CLK} = 32.768kHz	—	—	1	ms
FLL activation time (Fast activation)	T_{FLL2}	f_{CLK} = 32.768kHz	—	75	—	µs



●AC Characteristics (Reset)

(V_{DD}=1.7 to 1.9V, GND= 0V, Ta=-40 to 85°C)

Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
Reset pulse width	P_{RST}	—	400	—	—	µs
Reset noise elimination pulse width	P_{NRST}	—	—	—	0.1	

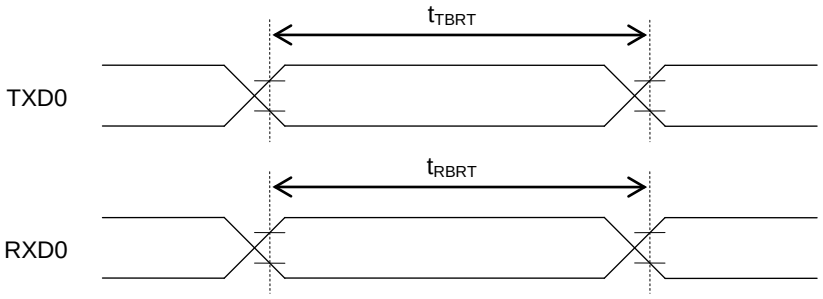


●AC Characteristics (UART)

(V_{DD}=1.7 to 1.9V, GND=0V, Ta=-40 to 85°C)

Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
Transferring baud-rate	t_{TBRT}	—	—	BRT*1	—	s
Receiving baud-rate	t_{RBRT}	—	BRT*1 -3%	BRT*1	BRT*1 +3%	s

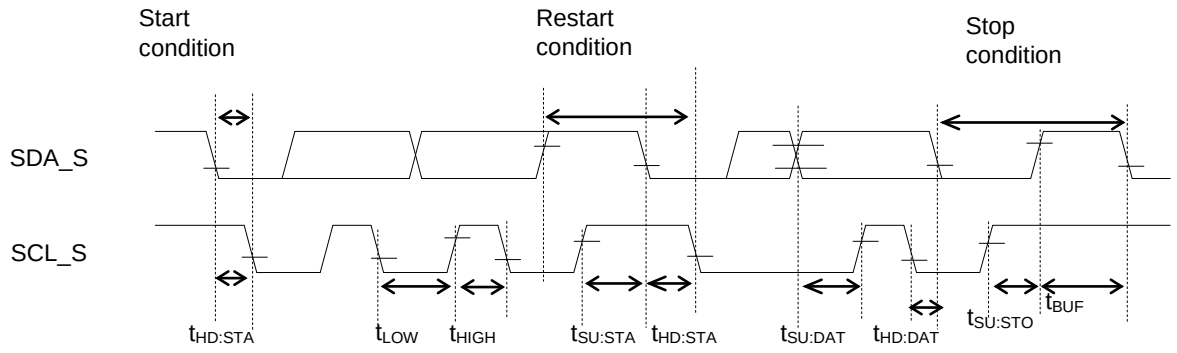
*1 Baud rate period (including the error of the clock frequency selected) set with the UART0 baud rate register (UA0BRTL,H) and the UART0 mode register 0 (UA0MOD0).



●AC Characteristics (Host Interface: I2C Slave Interface)

(V_{DD}=1.7 to 1.9V, GND=0V, Ta= -40 to 85°C)

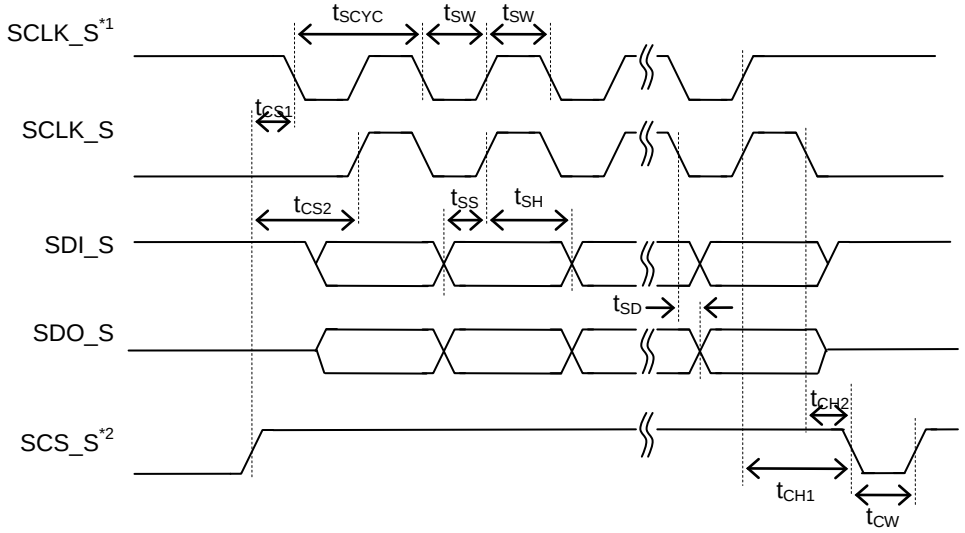
Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
SCL_S clock frequency	f_{SCL}	—	—	—	400	kHz
SCL_S hold time (start/restart condition)	$t_{HD:STA}$	—	0.6	—	—	µs
SCL_S "L" level time	t_{LOW}	—	1.3	—	—	µs
SCL_S "H" level time	t_{HIGH}	—	0.6	—	—	µs
SCL_S setup time (restart condition)	$t_{SU:STA}$	—	0.6	—	—	µs
SDA_S hold time	$t_{HD:DAT}$	—	0	—	—	ns
SDA_S setup time	$t_{SU:DAT}$	—	0.1	—	—	µs
SDA_S setup time (P: Stop condition)	$t_{SU:STO}$	—	0.6	—	—	µs
Bus free time	t_{BUF}	—	1.3	—	—	µs



●AC Characteristics (Host Interface: SPI Slave Interface)

(V_{DD}=1.7 to 1.9V, GND=0V, Ta= -40 to 85°C)

Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
SCLK_S input cycle	t_{SCYC}	—	250	—	—	ns
SCLK_S input pulse width	t_{SW}	—	120	—	—	ns
SCS_S setup time	t_{CS1}	—	80	—	—	ns
	t_{CS2}	—	80	—	—	ns
SCS_S hold time	t_{CH1}	—	80	—	—	ns
	t_{CH2}	—	80	—	—	ns
SCS_S input pulse width	t_{CW}	—	90	—	—	ns
SDO_S output delay time	t_{SD}	—	—	—	100	ns
SDI_S input setup time	t_{SS}	—	60	—	—	ns
SDI_S input hold time	t_{SH}	—	60	—	—	ns



*1 As for SPI, if Host keeps SCLK high while communication is standby, there are cases when this LSI cannot enter SleepDeep mode until SCS_S becomes non-active.
*2 Either “High active” or “Low active” can be selected for polarity of SCS_S.

●AC Characteristics (I2C Master Interface: Standard Mode 100 kHz)

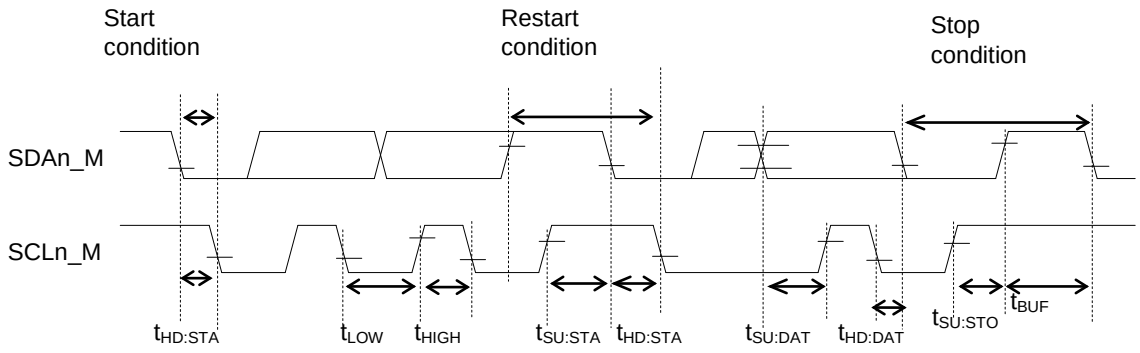
(V_{DD}=1.7 to 1.9V, GND=0V, Ta= -40 to 85°C)

Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
SCLn_M clock frequency	f_{SCL}	—	—	—	100	kHz
SCLn_M hold time (start/restart condition)	$t_{HD:STA}$	—	4.0	—	—	µs
SCLn_M "L" level time	t_{LOW}	—	4.7	—	—	µs
SCLn_M "H" level time	t_{HIGH}	—	4.0	—	—	µs
SCLn_M setup time (restart condition)	$t_{SU:STA}$	—	4.7	—	—	µs
SDAn_M hold time	$t_{HD:DAT}$	—	0	—	—	µs
SDAn_M setup time	$t_{SU:DAT}$	—	0.25	—	—	µs
SDAn_M setup time (P: Stop condition)	$t_{SU:STO}$	—	4.0	—	—	µs
Bus free time	t_{BUF}	—	4.7	—	—	µs

●AC Characteristics (I2C Master Interface: Fast Mode 400 kHz)

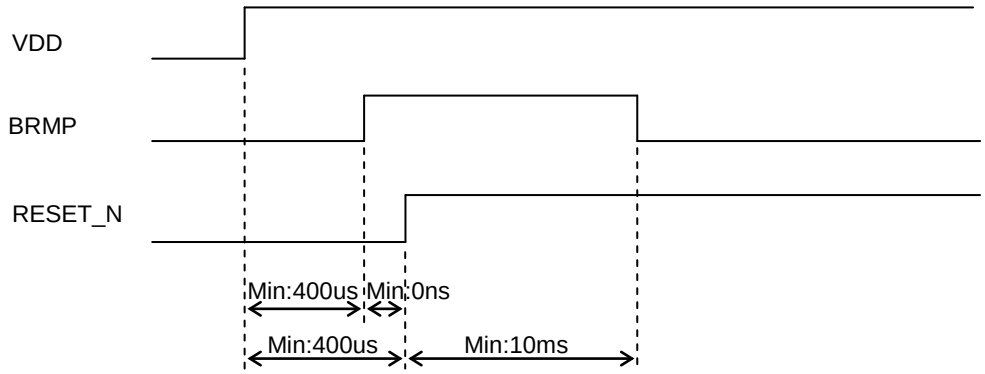
(V_{DD}=1.7 to 1.9V, GND=0V, Ta= -40 to 85°C)

Parameter	Symbol	Condition	Standard value			Unit
			Min.	Typ.	Max.	
SCLn_M clock frequency	f_{SCL}	—	—	—	400	kHz
SCLn_M hold time (start/restart condition)	$t_{HD:STA}$	—	0.6	—	—	µs
SCLn_M "L" level time	t_{LOW}	—	1.3	—	—	µs
SCLn_M "H" level time	t_{HIGH}	—	0.6	—	—	µs
SCLn_M setup time (restart condition)	$t_{SU:STA}$	—	0.6	—	—	µs
SDAn_M hold time	$t_{HD:DAT}$	—	0	—	—	µs
SDAn_M setup time	$t_{SU:DAT}$	—	0.1	—	—	µs
SDAn_M setup time (P: Stop condition)	$t_{SU:STO}$	—	0.6	—	—	µs
Bus free time	t_{BUF}	—	1.3	—	—	µs

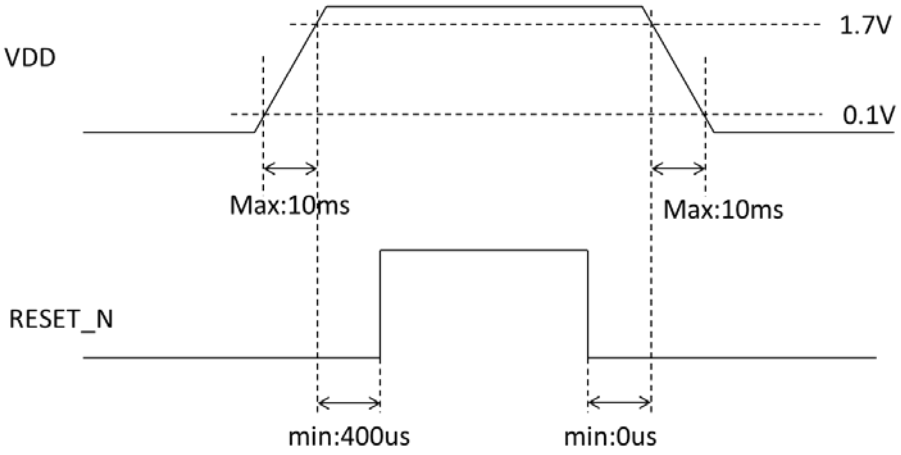


n=0, 1

●AC Characteristics (Firmware update)



●Power-on/Power-off



Appendix D Application Circuit Example

●Overview

Figure D-1 shows the examples of application circuits of ML630Q791, and Table D-1 shows the recommended values of capacitors and resistors in the circuits.

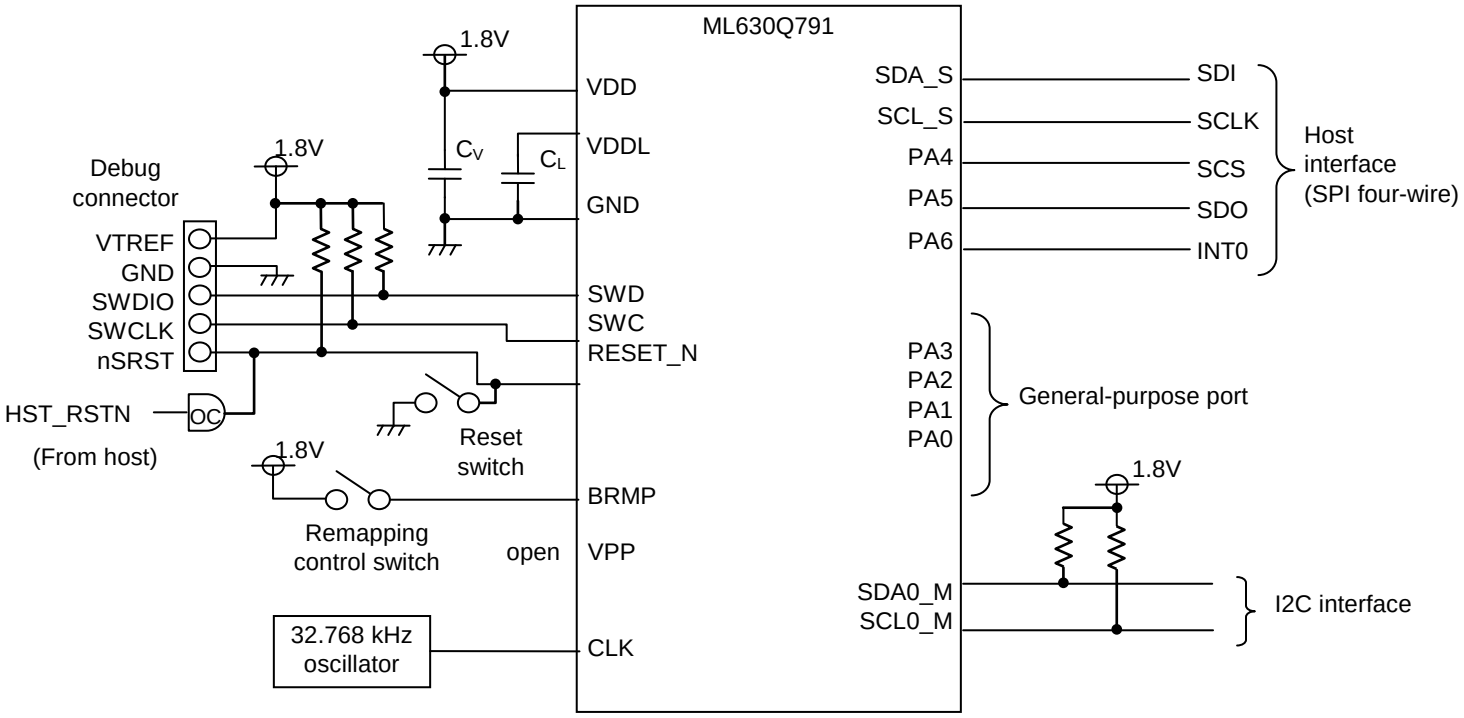


Figure D-1 Example of Application Circuit of ML630Q791

Table D-1 Recommended Values of Circuit Constants

Symbol	Recommended value
C_V	1.0uF
C_L	2.2uF

●Host Interface

With ML630Q791, the interface with the host processor can be selected from either I2C or SPI serial interface by setting the HIFCFG register. Connection example for each is shown below.

●I2C Slave Interface

To use an I2C interface, set the IFSEL bit of HIFCFG register to “1”. For ML630Q791, an I2C slave address can be set by the HIFCFG register, and the initial value is 17H. Figure D-2 shows an I2C connection example.

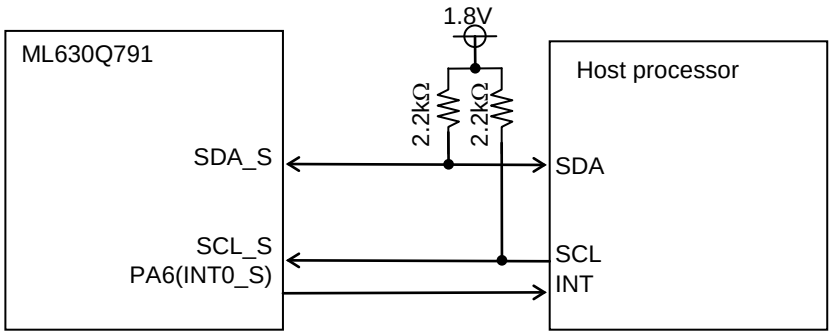


Figure D-2 Example of Connection with the Host Processor Using I2C

●SPI Slave Interface

To use an SPI interface, set the IFSEL bit of HIFCFG register to "0".

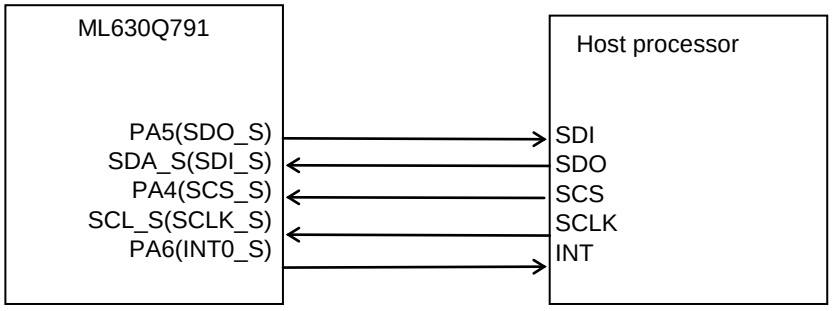


Figure D-3 Example of Connection with the Host Processor Using SPI

●Sensor Interface

ML630Q791 has 2 channels of I2C master interface for sensor control. The following shows a ch0 connection example.

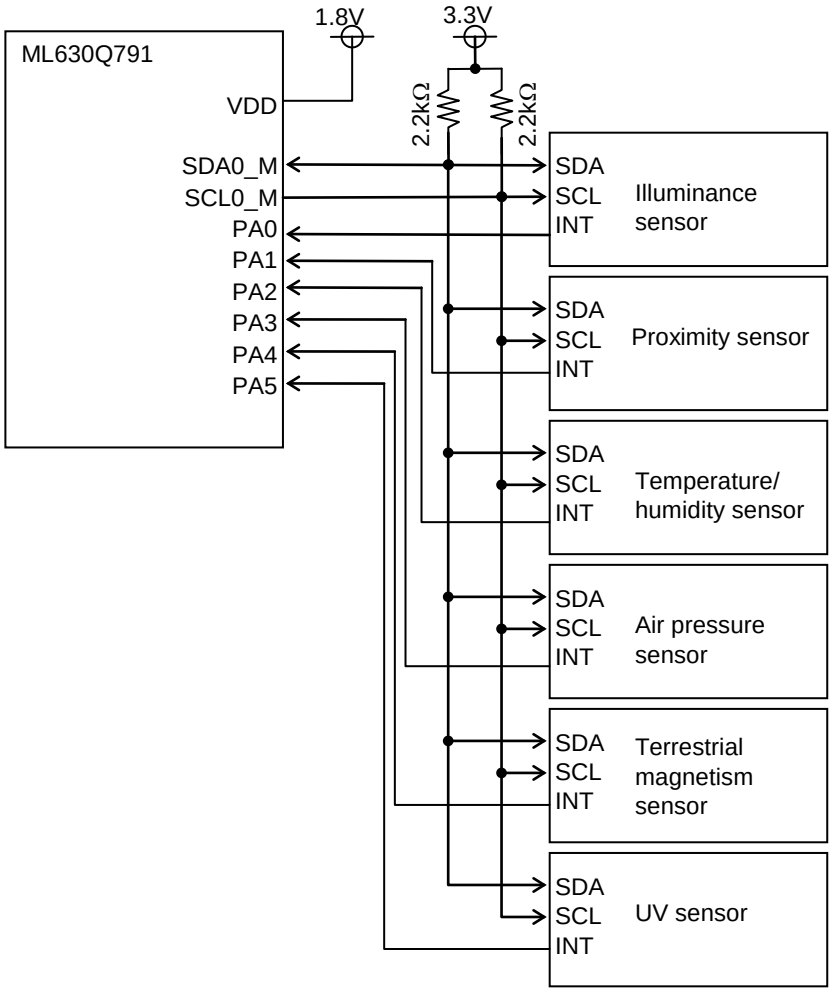


Figure D-4 Example of Connection with Sensor IC Using I2C

●Debug Interface

ML630Q791 has SW-DP as the debug interface.
For details of SW-DP, see "Cortex-M0 Technical Reference Manual" and other references.

●Firmware Update

With ML630Q791, remapping to the dedicated boot loader program is required to update the firmware. The remapping can be performed in one of two ways: through software (software remap), or forcing it by entering an appropriate sequence to the external pin (hardware remap). Figure D-5 shows circuit examples for the hardware remap. The additional circuit is not necessary for the software remap.

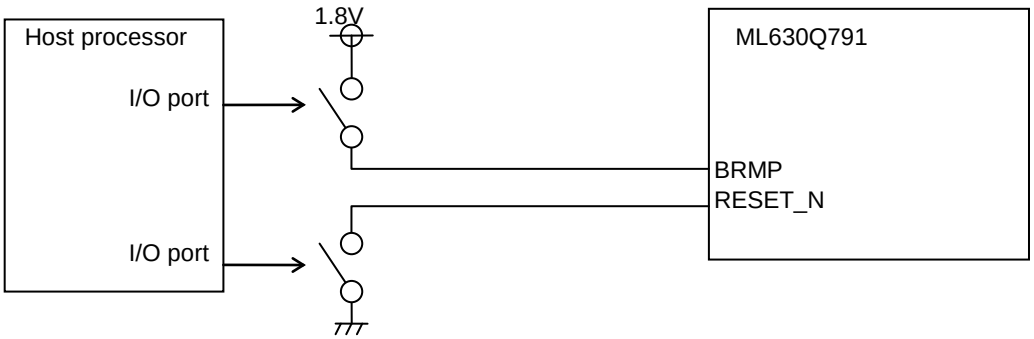


Figure D-5 Example of Circuit to Perform Firmware Update by External Pin

To begin the remapping sequence using the external pin, enter “L” level to the RESET_N pin. Next, after enter “H” level to the BRMP pin, enter "H" level to the RESET_N pin. Based on the BRMP pin that is set during this reset period, the area containing boot loader is re-mapped to reset vector. For details of the remapping timing, see Appendix C “Electrical Characteristics”.

●Clock Input Circuit

ML630Q791 inputs an external 32.768 kHz clock to the CLK pin.

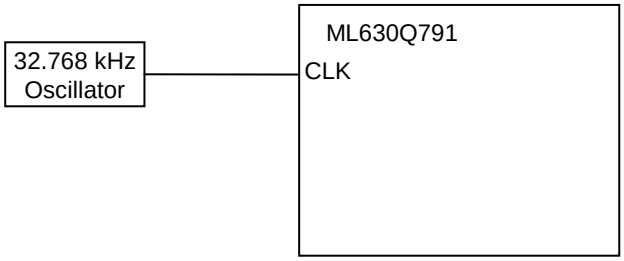


Figure D-6 External Clock Input Pin

Revision History

Revision History

Document No.	Date	Page		Description
		Previous Edition	Current Edition	
PEUL630Q791-01	July. 23. 2014	–	–	Preliminary Edition issued
FEUL630Q791-01	Oct. 22. 2014	–	–	Final Edition issued